O'REILLY®

Второе издание

Linux
книга рецептов

Все необходимое для администраторов и пользователей

Rescuer



Карла Шрёдер

SECOND EDITION

Linux Cookbook

Essential Skills for Linux Users and System and Network Administrators

Carla Schroder

Beijing • Boston • Farnham • Sebastopol • Tokyo



Linux
книга рецептов

Все необходимое для администраторов и пользователей

2-е издание

Карла Шрёдер



Санкт-Петербург • Москва • Минск

2022

ББК 32.973.2-018.2
УДК 004.451
Ш85

Шрёдер Карла

Ш85 Linux. Книга рецептов. 2-е изд. — СПб.: Питер, 2022. — 592 с.: ил. — (Серия «Бестселлеры O'Reilly»).

ISBN 978-5-4461-1937-0

Книга рецептов обучит начинающих пользователей и администраторов Linux управлять системой, используя как графические инструменты, так и командную строку. Независимо от того, используете ли вы Linux во встроенных или настольных системах, серверах, облачных или виртуальных средах, фундаментальные приемы одни. Цель книги — помочь вам быстро приступить к работе на простых и наглядных примерах. Карла Шрёдер приводит рецепты с объяснениями для конкретных ситуаций, а также ссылки для дополнительного изучения.

16+ (В соответствии с Федеральным законом от 29 декабря 2010 г. № 436-ФЗ.)

ББК 32.973.2-018.2
УДК 004.451

Права на издание получены по соглашению с O'Reilly. Все права защищены. Никакая часть данной книги не может быть воспроизведена в какой бы то ни было форме без письменного разрешения владельцев авторских прав.

Информация, содержащаяся в данной книге, получена из источников, рассматриваемых издательством как надежные. Тем не менее, имея в виду возможные человеческие или технические ошибки, издательство не может гарантировать абсолютную точность и полноту приводимых сведений и не несет ответственности за возможные ошибки, связанные с использованием книги. Издательство не несет ответственности за доступность материалов, ссылки на которые вы можете найти в этой книге. На момент подготовки книги к изданию все ссылки на интернет-ресурсы были действующими.

ISBN 978-1492087168 англ.
Authorized Russian translation of the English edition of Linux Cookbook, 2nd Edition ISBN 9781492087168 © 2021 Carla Schroder
This translation is published and sold by permission of O'Reilly Media, Inc., which owns or controls all rights to publish and sell the same.
ISBN 978-5-4461-1937-0
© Перевод на русский язык ООО «Прогресс книга», 2022
© Издание на русском языке, оформление ООО «Прогресс книга», 2022
© Серия «Бестселлеры O'Reilly», 2022

Краткое содержание

Оглавление

Предисловие

Давным-давно я написала первое издание этой книги. Оно увидело свет в 2004 году[1]. Книга хорошо продавалась, я познакомилась со многими счастливыми читателями, а некоторые даже стали моими друзьями.

Для книги о Linux 17 лет — большой срок. В 2004 году операционной системе Linux исполнилось 14 лет. Это была скромная операционная система для детских компьютеров. Но, несмотря на это, она уже пользовалась популярностью: ее активно устанавливали как на крошечные встраиваемые устройства, так и на мейнфреймы и суперкомпьютеры. Быстрое развитие Linux отчасти объясняется тем, что она является бесплатным клоном Unix, самой зрелой и эффективной операционной системы из всех существующих. Другой важный фактор быстрого развития и роста популярности Linux — отсутствие препятствий. Каждый может скачать эту операционную систему и попробовать ее в деле, а исходный код доступен всем, кто захочет внести в его развитие свой вклад.

В то время эта ОС была отличным примером действия принципа «форма следует за функцией». Она работала, была надежной, но неказистой, и, чтобы получить удобную для работы среду, пользователям приходилось подкручивать множество разных настроек то тут, то там. В ту пору для запуска системы Linux нужно было уметь обращаться с множеством команд, скриптов и файлов конфигурации, а также немножечко колдовать. Управление программным обеспечением, хранилищем, настройка сети, аудио, видео, управление ядром, процессами... все это требовало практической работы и постоянного изучения.

Семнадцать лет спустя все важные подсистемы Linux существенно изменились и улучшились. Теперь все настройки, которые раньше приходилось выполнять вручную, заменены тем, что я называю «просто работает». Пользоваться Linux стало гораздо проще, и мы можем сосредоточиться на своей работе вместо того, чтобы плясать с бубном и произносить заклинания во имя функционирования системы.

[1] В России первое издание вышло в 2006 году. — *Здесь и далее примеч. пер.*

Я рада представить это обновленное второе издание «Linux. Книга рецептов» и надеюсь, что вам понравится узнавать обо всех замечательных новшествах.

Для кого эта книга

Книга предназначена для читателей, имеющих некоторый опыт работы с компьютером, но не обязательно с Linux. Я постаралась сделать книгу максимально доступной для новичков в Linux. И тем не менее желательно, чтобы вы понимали базовые сетевые концепции, такие как IP-адресация, Ethernet, Wi-Fi, клиент и сервер, знали основы устройства компьютерного оборудования и имели некое представление об использовании командной строки. Если вы нуждаетесь в объяснении этих концепций, то я рекомендую обратиться к другим специализированным ресурсам; не поймите меня неправильно, я просто не хотела увязнуть в учебном материале, который уже хорошо документирован.

Все рецепты выработаны на практике. Я хотела бы, чтобы вы добивались успеха с первой попытки и не особенно расстраивались, если что-то не получится. Универсальный компьютер с Linux — сложная машина, требующая изучения и освоения. Будьте терпеливы, не торопитесь и читайте больше, чем хотелось бы. Часто нужные ответы оказываются в паре предложений от того места, где вы закончили читать.

Каждый дистрибутив Linux имеет встроенную документацию по командам, которая называется *man-страницами* (сокращенно от англ. manual pages — «страницы справочного руководства»). Например, `man 1 ls` покажет описание команды `ls`, которая выводит список содержимого каталога. Вводите эти команды точно так, как показано в книге, чтобы открыть правильную страницу руководства. Эту же информацию можно найти в Интернете.

Почему я написала эту книгу

Я давно хотела написать такую книгу, в которой собраны самые необходимые, на мой взгляд, рецепты работы с Linux. Эта система повсюду, и где бы вы ни столкнулись с ней, Linux — это Linux и навыки работы с ней универсальны. Мир технологий быстро развивается, и я думаю, что книга обеспечит прочную основу, на которую вы сможете опереться, независимо от направления ваших интересов.

Формат сборника рецептов особенно хорош для обучения основам, поскольку показывает, как решать конкретные реальные проблемы, и отделяет многословные объяснения от практических шагов, необходимых для решения задачи.

Структура издания

Книга не является официальным учебным руководством, которое нужно читать по порядку, от начала и до конца. Вы можете читать ее в любом порядке и искать то, что вам нужно.

Вот примерное содержание.

- Главы 1, 2 и 3 описывают установку Linux, управление загрузчиком, остановку и запуск системы, а также включают ответы на вопросы «Где взять Linux и как ее запустить».
- Глава 4 содержит введение в управление службами с помощью systemd, что является большим усовершенствованием по сравнению со старым способом изучения всевозможных сценариев, конфигурационных файлов и команд.
- Глава 5 рассказывает об управлении пользователями и группами, глава 6 — об управлении файлами и каталогами, а глава 7 — о резервном копировании и восстановлении. Эти три главы особенно важны для поддержки работоспособности и безопасности системы.
- Главы 8, 9 и 11 посвящены дисковым разделам и файловым системам (ФС), которые имеют фундаментальное значение для управления хранилищами данных. Управление данными — самый важный аспект вычислений.
- Глава 10 одна из самых увлекательных. Она поможет вам получить подробную информацию об аппаратном обеспечении вашего компьютера, не вскрывая корпус. Современное оборудование персональных компьютеров само сообщает массу интересной информации о себе, и Linux дополняет эту информацию своими сведениями.
- Главы 12 и 13 рассказывают о настройке безопасного удаленного доступа, а глава 14 описывает превосходный firewalld — динамический брандмауэр, который легко справляется с такими сложнейшими сценариями, как маршрутизация между разными сетями и управление несколькими сетевыми интерфейсами.
- Глава 15 знакомит с новыми возможностями CUPS (Common Unix Printing System) — универсальной системы печати Unix, — включая «печать без

драйверов», которая особенно востребована для мобильных устройств, поскольку позволяет подключать их к принтеру, избегая скачивания большого количества программного обеспечения.

- Глава 16 показывает, как управлять собственными службами сетевых имен с помощью превосходного Dnsmasq. Благодаря поддержке новых протоколов Dnsmasq сохранил свою актуальность, а старые команды и параметры конфигурации не изменились. Это первоклассный сервер имен, который легко интегрирует DNS и DHCP для централизованного управления IP-адресацией и специализированными сетевыми службами.
- Глава 17 знакомит с chrony и timesyncd, двумя новыми реализациями протокола сетевого времени (Network Time Protocol, NTP), а также с проверенными временем сервером и клиентом ntp.
- Глава 18 рассказывает об установке Linux на Raspberry Pi — популярный, небольшой и недорогой одноплатный компьютер — и его использовании для создания брандмауэра/шлюза в Интернете.
- Глава 19 показывает, как использовать SystemRescue для сброса утерянных паролей Linux и Windows, восстановления незагружающихся систем, восстановления данных в неисправной системе и описывает настройки SystemRescue, чтобы сделать этот инструмент еще полезнее.
- Главы 20 и 21 рассказывают об основных способах устранения неполадок с упором на поиск файлов журналов, зондирование сетей, а также зондирование и мониторинг оборудования.
- Приложение содержит шпаргалки с подсказками по управлению установкой и обслуживанием программного обеспечения.

Условные обозначения

В этой книге используются следующие условные обозначения.

Курсив

Курсивом выделены новые термины.

`Моноширинный шрифт`

Используется для листингов программ, а также внутри абзацев для обозначения таких элементов, как переменные и функции, базы данных, типы данных, переменные среды, операторы и ключевые слова, имена файлов и их расширений, названия каталогов.

`Моноширинный жирный шрифт`

Показывает команды или другой текст, который пользователь должен ввести самостоятельно.

`Моноширинный курсив`

Показывает текст, который должен быть заменен значениями, введенными пользователем, или значениями, определяемыми контекстом.

Шрифт без засечек

Используется для обозначения URL, адресов электронной почты, названий кнопок, названий клавиш и их сочетаний, элементов интерфейса.



Этот рисунок указывает на совет или предложение.



Этот рисунок указывает на общее замечание.



Этот рисунок указывает на предупреждение.

Использование программного кода примеров

Данная книга должна помочь решить ваши задачи. В общем случае все примеры кода из нее вы можете использовать в своих программах и в документации. Вам не нужно обращаться в издательство за разрешением, если вы не собираетесь воспроизводить существенные части кода. Например, если вы разрабатываете программу и используете в ней несколько отрывков кода из книги, то вам не нужно обращаться в издательство O'Reilly за разрешением. Однако вам необходимо его получить в случае продажи или распространения примеров из этой книги. Если вы отвечаете на вопросы, цитируя издание или примеры из него, то получать разрешение издательства не требуется. Но вам необходимо

будет его получить при включении существенных объемов программного кода примеров из этой книги в вашу документацию.

Мы приветствуем, но не требуем добавлять ссылку на первоисточник при цитировании. Под ссылкой на первоисточник мы подразумеваем указание авторов, издательства и ISBN. Например: «Linux. Книга рецептов. Второе издание. Карла Шрёдер (Питер). Copyright 2021 Carla Schroder, 978-5-4461-1937-0».

Если вам покажется, что использование кода примеров выходит за рамки оговоренных выше условий и разрешений, свяжитесь с нами по адресу permissions@oreilly.com.

Благодарности

Мне очень повезло с этой книгой. Мой редактор Джефф Блейл (Jeff Bleiel) неизменно поддерживал меня и во всем помогал. Он сам внес множество улучшений и организовывал поступательное развитие проекта.

Мой стажер Кейт Урнесс (Kate Urness) почти ничего не знала о Linux в начале этого приключения, что сделало ее идеальным рецензентом. Она протестировала каждый рецепт и помогла повысить их точность и ясность. Мы вместе выпили галлоны кофе и повеселились от души, что тоже было существенным вкладом.

Научный редактор Дэниел Барретт (Daniel Barrett) проявил потрясающее внимание к деталям и неустанно стремился к большей точности формулировок и описаний. Он тоже внес множество улучшений. Написать книгу, наполненную командами, — просто, а объяснить, как они работают, — сложно. Пусть каждому писателю повезет так, как мне, и у них будут такие же технические редакторы.

Научный редактор Джонатан Джонсон (Jonathan Johnson) нашел то, что упустили все остальные, добавил несколько суперкрутых заклинаний, привнес толику юмора и любезно разрешил мне сказать вам, что был очень нужен мне.

Зан Маккуэйд (Zan McQuade), представитель отдела закупок, затеял все это безумие. На протяжении многих лет он вел разговоры об обновлении данной книги и наконец добился своего.

Нескончаемый поток благодарностей моему супругу Терри, который кормил мулов, кошек, собак и меня, воодушевлял меня и удерживал от побега из дома, поскольку я потеряла рассудок и согласилась написать еще одну книгу.

Особая благодарность нашим котам и кошкам: Герцогине (рис. 1), Хороняке (рис. 2) и Безумному Максу (рис. 3), которые постоянно появляются в этой книге. Они очень помогли тем, что спали на моей клавиатуре, не позволяли мне сидеть на стуле и часто производили загадочные громкие звуки падения разных предметов.



Рис. 1. Герцогиня держит меня за ноги для моего же блага

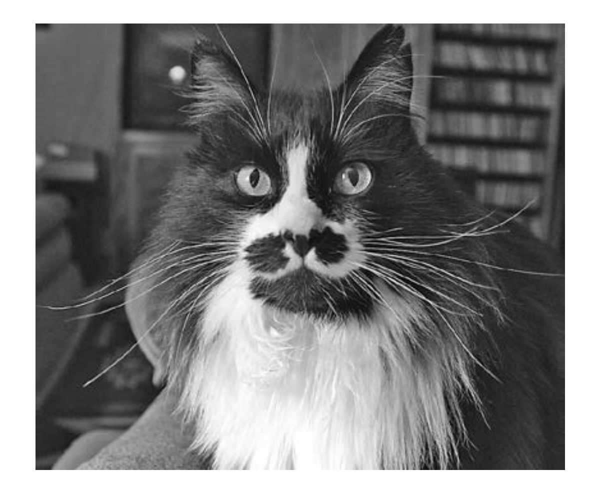


Рис. 2. Хороняка — наш гламурный мальчик



Рис. 3. Безумный Макс отдыхает перед следующим погромом

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу comp@piter.com (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

На веб-сайте издательства www.piter.com вы найдете подробную информацию о наших книгах.

ГЛАВА 1

Установка Linux

Первое препятствие, с которым сталкиваются новые пользователи Linux, — ее установка. Linux — самая простая в установке операционная система: вставьте установочный диск, ответьте на несколько вопросов, а затем займитесь какими-нибудь другими делами, пока процесс не завершится. В данной главе вы узнаете, как установить Linux, запустить ее live-версию, настроить мультизагрузку нескольких дистрибутивов Linux на одном компьютере и двухвариантную загрузку с Microsoft Windows.



Эксперименты с Linux

Чтобы обрести уверенность, нужна свобода совершать ошибки, поэтому по возможности используйте для знакомства с Linux второй компьютер. Если это невозможно, то своевременно создавайте резервные копии своих данных. Разрушенную систему Linux можно восстановить, но ваши данные незаменимы. Если вы настраиваете двухвариантную загрузку с Windows, то убедитесь, что у вас есть носитель с установочной версией Windows и носитель для восстановления.

Большинство дистрибутивов Linux предоставляют установочные образы двойного назначения: их можно запускать прямо с накопителя USB (live-версии) и устанавливать на жесткий диск. Когда система запускается прямо с накопителя USB, то она не вносит никаких изменений в ваш компьютер — просто загрузите ее, проверьте и перезагрузите компьютер. Некоторые live-версии Linux, такие как Ubuntu, поддерживают хранение данных на USB-накопителе, что позволяет создать носимую версию Linux, которую можно запускать на любом компьютере.

Мультизагрузка — это установка на компьютер нескольких операционных систем с последующим выбором в меню загрузки той, которую вы хотите использовать.

Вы можете организовать мультизагрузку любой системы Linux, любой из бесплатных версий Unix (FreeBSD, NetBSD, OpenBSD), а также мультизагрузку Linux и Microsoft Windows. Установка Linux параллельно с Windows — распространенный способ знакомства с Linux для пользователей Windows, а также для пользователей, которым необходимы обе системы.

Вы спросите, а как насчет macOS компании Apple? Сожалею, но организовать мультизагрузку Linux и macOS довольно сложно, и с каждой новой версией macOS становится все сложнее. Альтернативой установке обеих систем на одной машине является запуск Linux в Parallels — виртуальной машине для macOS.

Вместо того чтобы устанавливать Linux самостоятельно, можно купить ПК с уже установленной ОС. Есть несколько неплохих производителей, продающих ноутбуки, настольные компьютеры и серверы с Linux, в том числе: System76, ZaReason, Linux Certified, Think Penguin, Entroware и Tuxedo Computers. Компания Dell расширяет свою линейку компьютеров с Linux, а поставщики корпоративных версий Linux — Red Hat, SUSE и Ubuntu — тесно сотрудничают с поставщиками оборудования, включая Dell, Hewlett-Packard и IBM.

Тем не менее умение устанавливать Linux — полезный навык. Он открывает целый мир экспериментов с настройками и способами аварийного восстановления. *Переключение между дистрибутивами* — старинная забава, когда вы скачиваете и пробуете разные дистрибутивы Linux.

Несмотря на то что для установки Linux требуется всего несколько шагов, вам понадобится определенный багаж знаний, особенно если вы хотите настроить свою установку, например распределить разделы диска определенным образом или организовать мультизагрузку с другими дистрибутивами Linux или с Microsoft Windows. Вам нужно знать, как войти в настройки базовой системы ввода/вывода (Basic Input Output System, BIOS) или единого расширяемого микропрограммного интерфейса (Unified Extensible Firmware Interface, UEFI). Вам нужен хороший доступ в Интернет. Все дистрибутивы Linux можно скачать бесплатно, даже коммерческие корпоративные дистрибутивы, такие как Red Hat, SUSE и Ubuntu. Размеры скачиваемых файлов варьируются от нескольких мегабайтов для сверхмалых дистрибутивов Linux, таких как Tiny Core Linux, который вмещает полную операционную систему с графическим рабочим столом в 12 Мбайт, до 10+ Гбайт для SUSE Linux Enterprise Server. Для большинства дистрибутивов Linux имеются установочные образы размером 2–4 Гбайт, которые идеально помещаются на DVD или небольшой USB-накопитель.

Большинство дистрибутивов Linux предоставляют образ для установки по сети; например, такой образ для Debian занимает около 200 Мбайт. Он содержит часть

системы Debian, которой вполне достаточно для того, чтобы загрузиться, подключиться к Интернету и скачать только необходимые пакеты вместо полного установочного образа.

Вы можете свободно делиться с другими людьми любым скачанным дистрибутивом Linux.

Кроме того, есть возможность купить дистрибутивы Linux на DVD и USB-накопителях. Посетите Shop Linux Online (https://shoplinuxonline.com) и Linux Disc Online (https://linuxdisconline.com), где предлагаются физические носители с установочными образами различных дистрибутивов Linux.

Загрузка с установочного носителя

Чтобы установить Linux, нужно загрузиться с установочного носителя USB или DVD. Возможно, для этого вам придется войти в настройки BIOS или UEFI вашего компьютера, чтобы разрешить загрузку со съемного устройства. Некоторые компьютеры дают возможность выбрать альтернативное загрузочное устройство без входа в BIOS/UEFI; например, мой ноутбук отображает экран заставки, на котором перечислены все поддерживаемые сочетания клавиш: F2 или Delete для входа в настройки и F11 для входа в меню выбора альтернативного загрузочного устройства. В системах Dell можно нажать клавишу F12, чтобы открыть меню однократной загрузки. Все компьютеры разные, поэтому загляните в руководство по материнской плате, чтобы узнать, какие возможности она предлагает.

Вероятно, вам придется отключить безопасную загрузку (secure boot) в настройках UEFI, чтобы разрешить загрузку со съемного носителя. Fedora, openSUSE и Ubuntu имеют собственные подписанные ключи и будут загружаться с включенной поддержкой безопасной загрузки. Другие дистрибутивы Linux, такие как SystemRescue (см. главу 19), не имеют подобных ключей.



Безопасная загрузка

Безопасная загрузка (Secure Boot) — это функция безопасности UEFI. Когда она включена, UEFI позволяет загружать только операционные системы, имеющие специальные подписанные ключи. Идея состоит в том, чтобы предотвратить управление загрузчиком со стороны вредоносного кода.

Большинство дистрибутивов Linux не имеют подписанных ключей, поэтому для их загрузки необходимо отключить безопасную загрузку.

Где скачать Linux

Существуют сотни дистрибутивов Linux, и отличное место, где можно узнать об их существовании, — это DistroWatch.com (https://distrowatch.com), наиболее исчерпывающий ресурс о дистрибутивах Linux. DistroWatch публикует обзоры, подробную информацию и новости, а также список, включающий 100 самых популярных дистрибутивов.

Лучший дистрибутив Linux для новичков

Linux предлагает довольно много хорошего, возможно даже слишком много. Рецепты, представленные в этой книге, были протестированы в openSUSE, Fedora Linux и Ubuntu Linux. Эти три дистрибутива прекрасно зарекомендовали себя, пользуются большой популярностью и удобны в обслуживании. Они представляют три разных семейства Linux (см. приложение). Как мне кажется, для новичков идеально подойдет дистрибутив Ubuntu с его простым установщиком, хорошей документацией и обширным и доброжелательным сообществом пользователей.

Каждый дистрибутив Linux имеет отличия: разные инструменты установки программного обеспечения, разные настройки по умолчанию, разные местоположения файлов... но все они похожи в своей основе. Большая часть из того, что вы узнаете о любом конкретном дистрибутиве, применима ко всем из них.



Аппаратные архитектуры

Авторы практических руководств раньше исходили из того, что читатели используют оборудование с архитектурой x86. Однако с ростом популярности процессоров ARM ситуация изменилась. Современные дистрибутивы Linux поддерживают большое количество аппаратных архитектур, и в рецепте 10.11 вы узнаете, как определить свою архитектуру. У вас не получится случайно установить версию Linux не для своей архитектуры, поскольку процедура установки сразу же потерпит неудачу и выведет сообщение об ошибке, объясняющее причину.

Установочные образы Linux распространяются в формате ISO 9660 и имеют расширение `*.iso`, например `ubuntu-20.04.1-desktop-amd64.iso` для компьютеров с аппаратной архитектурой x86-64 и `ubuntu-20.04.1-live-server-arm64.iso` для компьютеров с аппаратной архитектурой ARM. Это сжатый архив, содержащий целую файловую систему и программу установки. В процессе установки он распаковывается, и вы можете видеть все файлы.

Первоначально формат `*.iso` предназначался для CD и DVD. Когда-то дистрибутивы Linux умещались на одном компакт-диске. (Более того, были времена,

когда они умещались на нескольких 3,5-дюймовых дискетах!) Современные дистрибутивы Linux в большинстве своем слишком велики и уже не умещаются на CD. USB-накопители идеально подходят для установки Linux, поскольку они недорогие, многоразовые и намного быстрее оптических носителей.

1.1. Вход в настройки BIOS/UEFI

Задача

Войти в настройки BIOS/UEFI.

Решение

Войдите в настройки BIOS/UEFI, нажав при запуске соответствующую клавишу Fn. В системах Dell, ASUS и Acer это обычно F2, а в Lenovo — F1. В других системах это может быть другая клавиша; например, в ряде систем используется Delete, поэтому загляните в документацию к своему компьютеру. Некоторые системы сообщают, какую клавишу нажать, на экране запуска. Часто трудно нажать клавишу в нужное время, поэтому начинайте и продолжайте нажимать ее сразу после нажатия кнопки питания, как если бы вы нажимали кнопку вызова лифта, чтобы он прибыл быстрее.

Настройки UEFI в каждом компьютере выглядят по-разному; например, компьютеры компании Lenovo имеют яркий и хорошо организованный интерфейс (рис. 1.1).

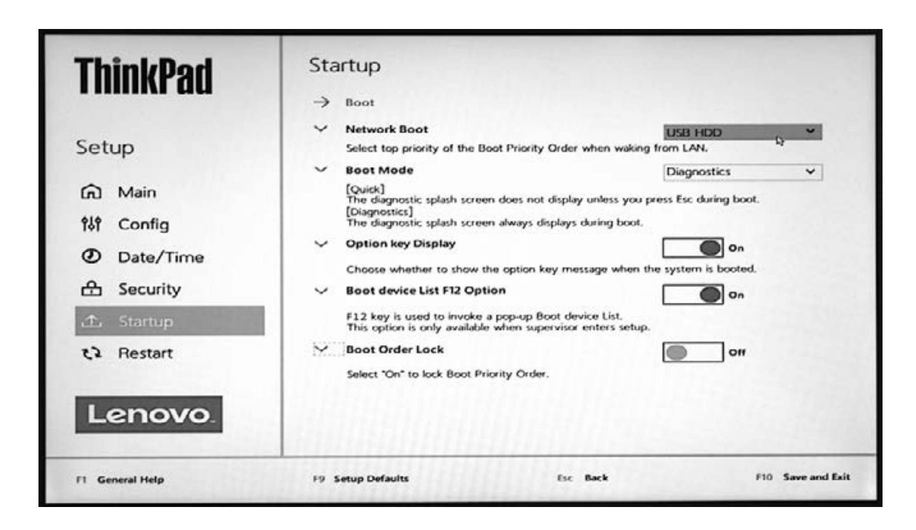


Рис. 1.1. Настройки UEFI в новом Lenovo ThinkPad

ASRock UEFI в моей тестовой системе выглядит мрачно и броско (рис. 1.2). Эта материнская плата предназначена для геймеров и имеет множество настроек для разгона процессора и других оптимизаций производительности. На рис. 1.2 показан браузер материнской платы; стоит навести указатель мыши на любой элемент, и вы тут же получите информацию о нем.

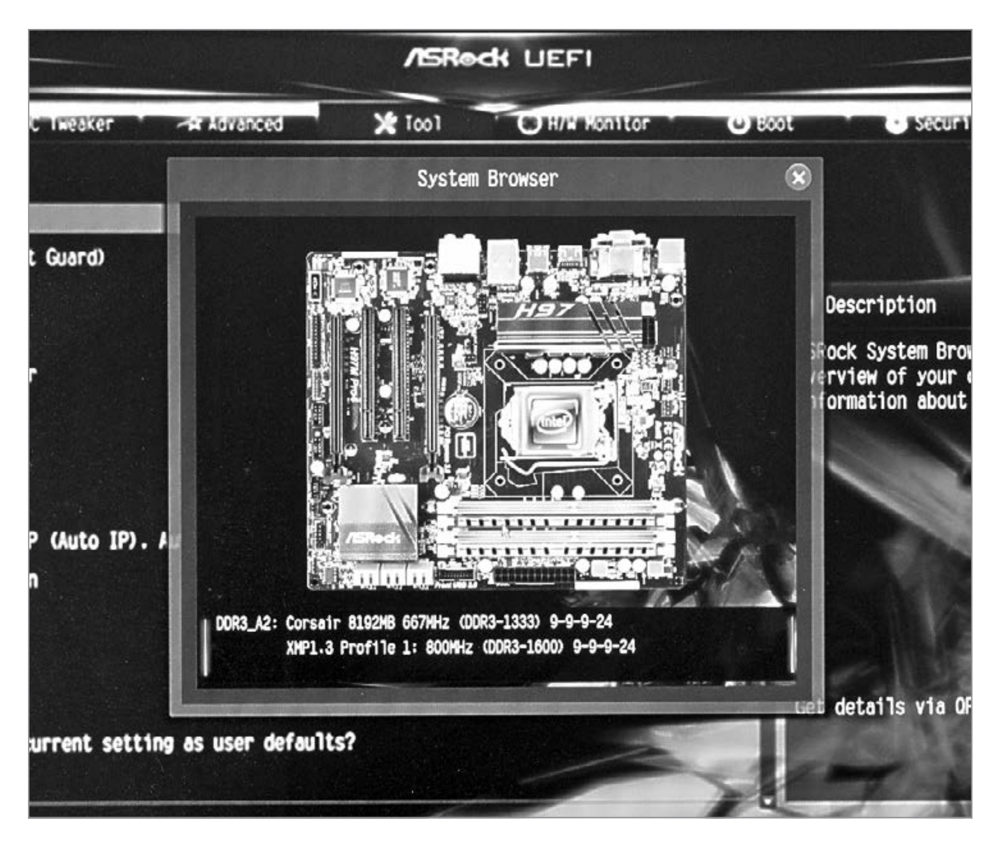


Рис. 1.2. Браузер материнской платы в ASRock UEFI

Комментарий

Когда вы загружаете компьютер, первые программные инструкции, которые он выполняет, извлекаются из прошивки BIOS или UEFI в материнской плате. BIOS — это старая система, используемая с 1980 года. UEFI — ее современная замена, которая включает поддержку устаревшей BIOS. Почти все компьютеры, выпущенные после середины 2000-х, имеют UEFI.

UEFI предлагает больше функций, чем старая BIOS, и напоминает небольшую операционную систему. Экраны настройки UEFI управляют порядком загрузки, загрузочными устройствами, параметрами безопасности, безопасной загрузкой, разгоном, отображением состояния оборудования и сети и предлагают множество других функций.

Дополнительная информация

- Документация для материнской платы.
- Форум Unified Extensible Firmware Interface (https://uefi.org).

1.2. Скачивание установочного образа Linux

Задача

Найти и скачать установочный образ Linux.

Решение

Прежде всего вы должны понять, какой дистрибутив Linux хотели бы попробовать. Если вы не знаете, с чего начать, то советую посмотреть в сторону Ubuntu Linux (https://ubuntu.com). Прекрасным выбором для новичков также станут Fedora Linux (https://getfedora.org) и openSUSE Linux (https://opensuse.org).

Когда скачивание завершится, образ желательно проверить. Этот важный шаг поможет вам убедиться, что образ не был поврежден в процессе скачивания и не был подменен где-то по пути.

Все производители дистрибутивов сопровождают свои установочные образы подписанными ключами и контрольными суммами. Ubuntu, например, предлагает инструкции с командами, которые можно просто копировать и вставлять в командную строку. Откройте терминал и перейдите в каталог, куда вы скачали образ с Ubuntu. Проверка образа Ubuntu 21.04 выглядит следующим образом:

```
$ echo "fa95fb748b34d470a7cfa5e3c1c8fa1163e2dc340cd5a60f7ece9dc963ecdf88 \
*ubuntu-21.04-desktop-amd64.iso" | shasum -a 256 --check

ubuntu-21.04-desktop-amd64.iso: OK
```

Если вы увидите сообщение shasum: WARNING: 1 computed checksum did NOT match (ВНИМАНИЕ: 1 вычисленная контрольная сумма НЕ совпадает), значит, образ

поврежден или вы скопировали не ту контрольную сумму. Чаще проблема связана с повреждением образа во время загрузки, поэтому скачайте его снова.

Другие дистрибутивы Linux предлагают немного иные методы проверки, поэтому следуйте инструкциям на их сайтах.

Комментарий

Существует отличный сайт Distrowatch.com (https://distrowatch.com), где можно познакомиться с сотнями дистрибутивов Linux. На Distrowatch публикуется больше новостей и информации о дистрибутивах Linux, чем где-либо еще.

Дополнительная информация

- `man 1 sha256sum`
- Ubuntu Linux (https://ubuntu.com).
- Fedora Linux (https://getfedora.org).
- openSUSE Linux (https://opensuse.org).

1.3. Создание загрузочного USB-накопителя с Linux с помощью UNetbootin

Задача

Вы скачали установочный iso-образ Linux, и теперь его нужно записать на USB-накопитель, чтобы создать загрузочный носитель. Вы предпочитаете инструменты с графическим инструментом.

Решение

Попробуйте UNetbootin (https://oreil.ly/8CXp9) — Universal Netboot Installer. Данный инструмент работает в Linux, macOS и Windows; с его помощью можно скачать и создать установочный диск Linux в любой из этих операционных систем. UNetbootin создает установочный USB-накопитель из скачанного образа `*.iso`, но также может скачать образ `*.iso` сам (рис. 1.3).

В качестве носителя может послужить USB-накопитель любой емкости (больше, чем размер файла `*.iso`, конечно). Образ `*.iso` займет устройство целиком, поэтому его не получится использовать для чего-то еще, и для каждого файла `*.iso` требуется отдельный USB-накопитель.

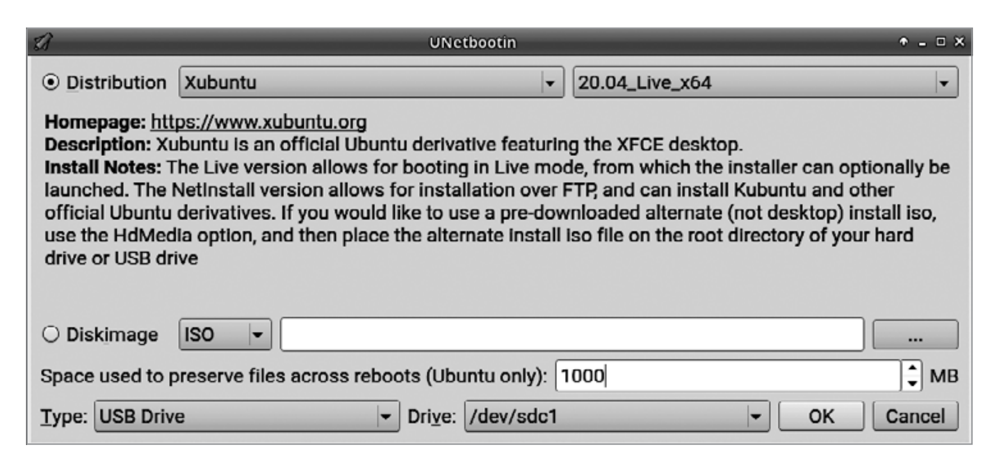


Рис. 1.3. Использование UNetbootin для создания установочного USB-накопителя с Linux

На сайте UNetbootin вы найдете все необходимые инструкции и файлы для скачивания. Некоторые дистрибутивы Linux предлагают свои пакеты с UNetbootin, но на сайте UNetbootin вы всегда найдете самые свежие версии.

Комментарий

В числе других интересных приложений с графическим интерфейсом можно назвать: USB Creator, ISO Image Writer и GNOME MultiWriter, последнее из которых может копировать образы сразу на несколько USB-накопителей.

Когда установочный USB-накопитель будет создан, вы можете просмотреть файлы на нем. Единственный образ `*.iso` разворачивается в целую файловую систему с множеством файлов и каталогов, как в следующем примере с Ubuntu:

```
$ ls -C1 /media/duchess/'Ubuntu 21.04.1 amd64'/
boot
casper
dists
EFI
install
isolinux
md5sum.txt
pics
pool
preseed
README.diskdefines
ubuntu
```

Каждый дистрибутив Linux включает свою программу установки. Вот, например, установочные файлы Fedora:

```
$ ls -C1 /media/duchess/Fedora-WS-Live-34-1-6/
EFI
images
isolinux
LiveOS
```

Было бы хорошо иметь один USB-накопитель с кучей установочных файлов Linux, и есть множество программ, помогающих создавать такие накопители. Я предпочитаю программу Ventoy (https://ventoy.net), которая поддерживает большое количество дистрибутивов Linux. Есть версии программы для Linux и Windows, и с их помощью можно создавать USB-накопители с установочными образами Linux для запуска live-версий и для установки на жесткие диски.

Дополнительная информация

- UNetbootin (https://oreil.ly/8CXp9).
- Глава 9.
- Ventoy (https://ventoy.net).

1.4. Создание установочного DVD с Linux с помощью K3b

Задача

Создать установочный DVD с Linux с помощью инструмента с графическим интерфейсом.

Решение

Используйте K3b (KDE Burn Baby Burn). K3b — это приложение для Linux с графическим интерфейсом для записи CD/DVD.

Если у вас нет системы Linux, то используйте любую другую программу записи образов ISO 9660 на CD/DVD. Выбранная вами программа должна поддерживать операцию, обозначаемую ею как «записать существующий образ на диск».

Запустив K3b, вы увидите окно, как показано на рис. 1.4. Нажмите кнопку Burn Image (Записать образ) и обратите внимание на подтверждение в левом нижнем углу — надпись Write an ISO 9660 ... image to an optical disk (Записать образ ISO 9660 ... на оптический диск).

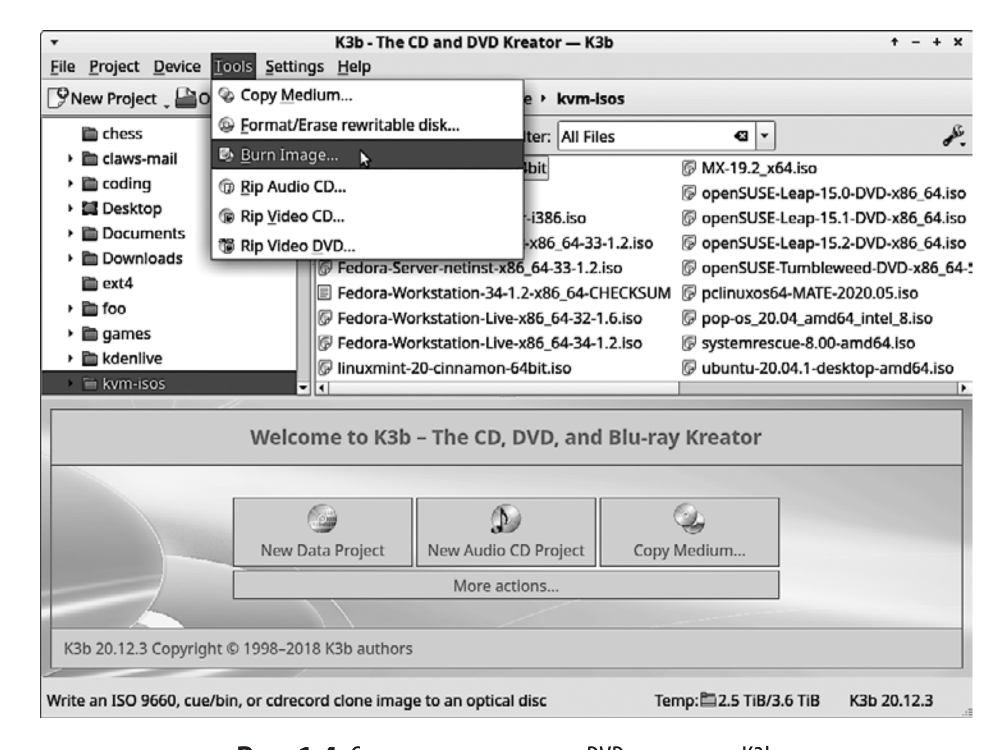


Рис. 1.4. Создание установочного DVD с помощью K3b

На следующем экране (рис. 1.5) в раскрывающемся списке вверху слева выберите свой образ `*.iso`. Затем вверху справа выберите ISO 9660 filesystem image (Образ файловой системы ISO 9660). Внизу на вкладке Settings (Настройки) установите флажок Verify written data (Проверить записанные данные). В этом случае программа вычислит контрольную сумму после записи образа и сравнит ее с контрольной суммой исходного образа `*.iso`. Это важный шаг, поскольку несовпавшие контрольные суммы означают, что диск поврежден и его нельзя использовать для установки.

Когда диск будет успешно записан, вы увидите сообщение о благополучном завершении, как показано на рис. 1.6. Если в процессе записи возникнут какие-либо ошибки, то на экране появятся полезные сообщения.

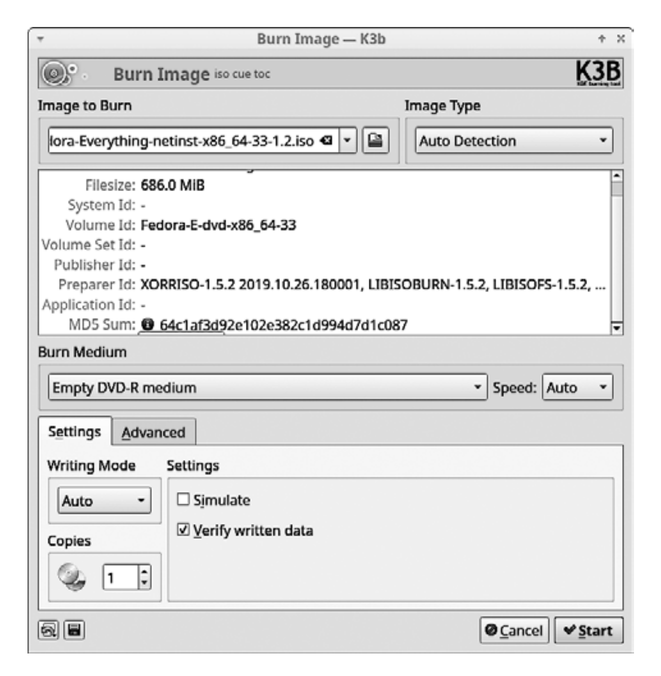


Рис. 1.5. Настройка записи

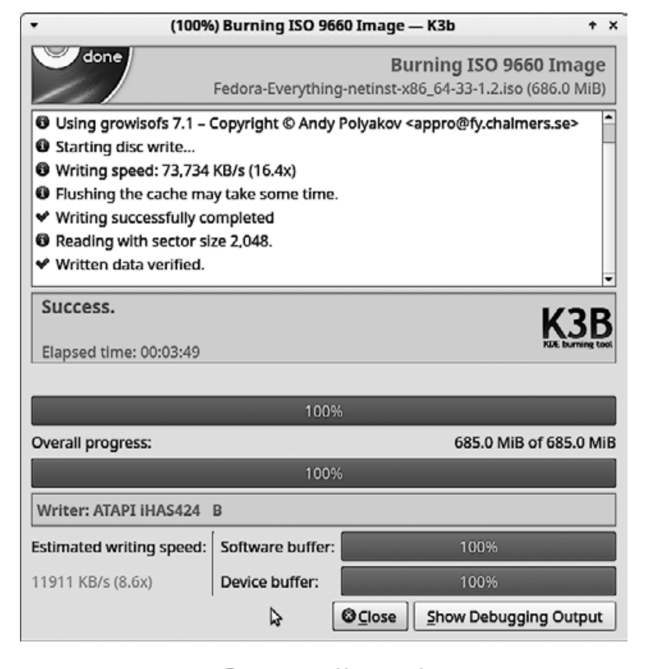


Рис. 1.6. Успешно!

Комментарий

Brasero и XFBurn — еще два отличных Linux-приложения для записи CD/DVD. Они имеют более простой интерфейс, чем K3b, но обладают достаточно широким набором функций.

Мир технологий быстро меняется. Всего несколько лет назад я часто пользовалась CD и DVD. Затем рынок захватили USB-накопители, и я уже много лет не записывала диски, пока не приступила к написанию этой главы.

В настоящее время производители стараются, чтобы CD и DVD ушли в небытие, отказываясь вставлять приводы CD/DVD в свои компьютеры. Однако это совсем не проблема, поскольку всегда можно приобрести внешний USB-привод CD/DVD. Можно даже найти приводы с питанием от шины, для подключения которых достаточно одного USB-кабеля, и нет необходимости возиться с кабелем питания. В настоящее время продолжают выпускаться болванки CD/DVD хорошего качества, так что если вы предпочитаете оптические диски, то они станут для вас надежным выбором.

Дополнительная информация

- K3b (https://oreil.ly/MJmXF).
- Brasero (https://oreil.ly/a9Dxx).

1.5. Создание загрузочного CD/DVD с помощью команды wodim

Задача

Создать загрузочный CD/DVD с помощью командной строки.

Решение

Попробуйте команду `wodim`. Обычно привод оптических дисков доступен через символическую ссылку `/dev/cdrom`, указывающую на `/dev/sr0`. Используйте символическую ссылку, поскольку для нее настроены правильные разрешения:

```
$ ls -l /dev | grep cdr
lrwxrwxrwx 1 root root          3 Mar  7 12:38 cdrom -> sr0
```

```
lrwxrwxrwx 1 root root          3 Mar  7 12:38 cdrw -> sr0
crw-rw----+ 1 root cdrom   21,  2 Mar  7 08:34 sg2
brw-rw----+ 1 root cdrom   11,  0 Mar  7 12:57 sr0
```

И скопируйте свой установочный образ на диск:

```
$ wodim dev=/dev/cdrom -v ubuntu-21.04-desktop-amd64.iso
```

Комментарий

В примере вывода команды `ls -l` выше можно видеть устройства sg2 и sr0. Устройство sg2 — это символьное устройство, а sr0 — блочное. Символьные устройства открывают прямой доступ к аппаратным устройствам через драйверы в ядре. Блочные — поддерживают буферизованный доступ к аппаратным устройствам через разные программы, которые выполняют операции чтения и записи с физическими носителями. Пользователи взаимодействуют с устройствами хранения, такими как DVD и жесткие диски, через драйверы блочных устройств в ядре. Список используемых модулей ядра, управляющих символьными и блочными устройствами, можно увидеть в файле `/boot/config-*`.

Дополнительная информация

- `man 1 wodim`

1.6. Создание установочного USB-носителя с Linux с помощью команды dd

Задача

Создать установочный USB-носитель с помощью командной строки, не используя инструменты с графическим интерфейсом.

Решение

Используйте команду `dd`, которая имеется во всех версиях Linux и везде работает одинаково.

Сначала определите имя устройства своего USB-накопителя с помощью команды `lsblk`, чтобы скопировать образ на правильное устройство. В моем случае, как показано ниже, USB-накопителем является устройство `/dev/sdb`:

```
$ lsblk -o NAME,FSTYPE,LABEL,MOUNTPOINT

NAME    FSTYPE LABEL       MOUNTPOINT
sda
├─sda1  vfat                /boot/efi
├─sda2  xfs    osuse15-2    /boot
├─sda3  xfs                 /
├─sda4  xfs                 /home
└─sda5  swap                [SWAP]
sdb
└─sdb1  xfs    32gbusb
sr0
```

Следующая команда создаст установочный USB-носитель и отобразит ход процесса копирования:

```
$ sudo dd status=progress if=ubuntu-20.04.1-LTS-desktop-amd64.iso of=/dev/sdb
211509760 bytes (212 MB, 202 MiB) copied, 63 s, 3.4 MB/s
```

Копирование займет несколько минут, и в конце будет выведено:

```
2782257664 bytes (2.8 GB, 2.6 GiB) copied, 484 s, 5.7 MB/s
5439488+0 records in
5439488+0 records out
2785017856 bytes (2.8 GB, 2.6 GiB) copied, 484.144 s, 5.8 MB/s
```

Извлеките накопитель, затем снова вставьте его в порт USB и посмотрите, какие файлы на нем хранятся. На рис. 1.7 представлено содержимое накопителя после записи установочного образа Ubuntu Linux, показанное в диспетчере файлов Thunar.

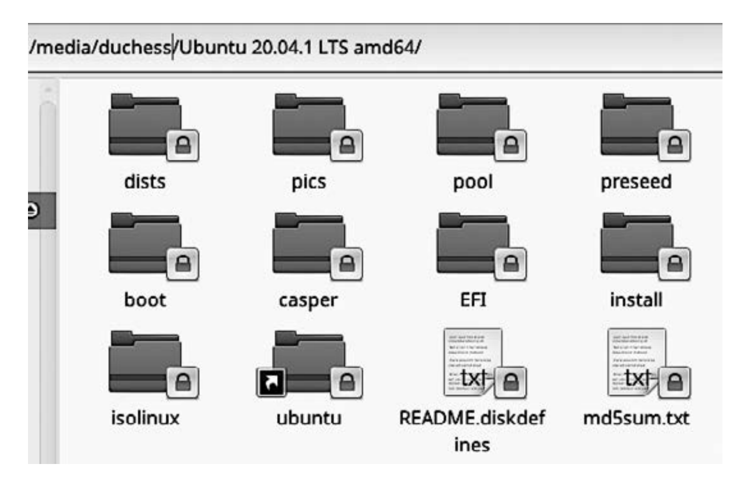


Рис. 1.7. Файлы из установочного образа Ubuntu Linux Ubuntu Linux, показанные в диспетчере файлов Thunar

Файлы отмечены значками закрытых замков, поскольку установщик Ubuntu использует файловую систему SquashFS, доступную только для чтения. Файлы можно читать, но нельзя удалять или изменять.

Установочный USB-накопитель готов к использованию.

Комментарий

Выбор правильного устройства для копирования установочных файлов — чрезвычайно важный шаг. В примере вывода команды `lsblk` выше присутствуют только два устройства хранения. Обратите внимание на столбец `LABEL` (Метка); вы можете добавлять метки к своим файловым системам, чтобы знать, какие это системы. (См. рецепт 9.2 и рецепты создания файловых систем в главе 11, чтобы узнать, как добавлять метки к файловым системам.)

Инструменты с графическим интерфейсом удобны, но я предпочитаю команду `dd`, поскольку она проста и надежна. Название `dd` возникло как сокращение от Disk Duplicator. Это одна из самых старых команд GNU, входящих в состав пакета GNU *coreutils*, который существует с момента возникновения Linux.

Дополнительная информация

- `man 1 dd`

1.7. Простая установка Ubuntu

Задача

Выполнить простую установку Ubuntu. Для этого у вас должен быть готов установочный носитель и вы должны знать, как с него загрузиться. На компьютере нет ничего, что требовалось бы сохранить, поэтому Ubuntu может занять весь жесткий диск.

Решение

Следующий пример демонстрирует быструю и простую установку Ubuntu Linux 21.04 (Hirsute Hippo[1]). Всем выпускам Ubuntu даются альтернативные названия животных.

---

[1] Шерстистый бегемот.

Вставьте установочный носитель, включите компьютер и откройте загрузочное меню в своей системе. Выберите загрузку с установочного носителя (рис. 1.8).

```
Use the ↑(Up) and ↓(Down) arrow keys to move the pointer to the desired boot device.
Press [Enter] to attempt the boot or ESC to Cancel. (* = Password Required)

Boot mode is set to: UEFI; Secure Boot: OFF

LEGACY BOOT:
    SAMSUNG SSD SM871 2.5 7mm
    ST1000DM003-1SB102
    PO: PLDS DVD+/-RW DH-16AES
UEFI BOOT:
    opensuse-secureboot
    Windows Boot Manager
    UEFI: ST1000DM003-1SB102
OTHER OPTIONS:
    BIOS Setup
    BIOS Flash Update
    Diagnostics
    Intel(R) Management Engine BIOS Extension (MEBx)
    Change Boot Mode Settings
```

Рис. 1.8. Загрузка с установочного USB-накопителя



Все меню UEFI выглядят по-разному

Экраны с настройками UEFI разных производителей выглядят по-разному. На снимке экрана, представленном выше, изображена страница с настройками UEFI в компьютере Dell.

Когда появится меню загрузчика GRUB, оставьте выбранным вариант по умолчанию. Для Ubuntu 21.04 это вариант Ubuntu (рис. 1.9).

```
GNU GRUB  version 2.04

*Ubuntu
 Ubuntu (safe graphics)
 OEM install (for manufacturers)
 Test memory

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands
   before booting or `c' for a command-line.
```

Рис. 1.9. Меню загрузчика GRUB установочного образа Ubuntu

Далее вам будет предложено выбрать один из двух вариантов: Try Ubuntu (Попробовать Ubuntu) и Install Ubuntu (Установить Ubuntu). При выборе первого варианта запустится live-версия Ubuntu, а если выбрать второй вариант, то откроется окно программы установки (рис. 1.10). Неважно, какой вариант вы выберете, поскольку при выборе первого варианта на рабочем столе останется большая кнопка, нажатие которой запускает установку.

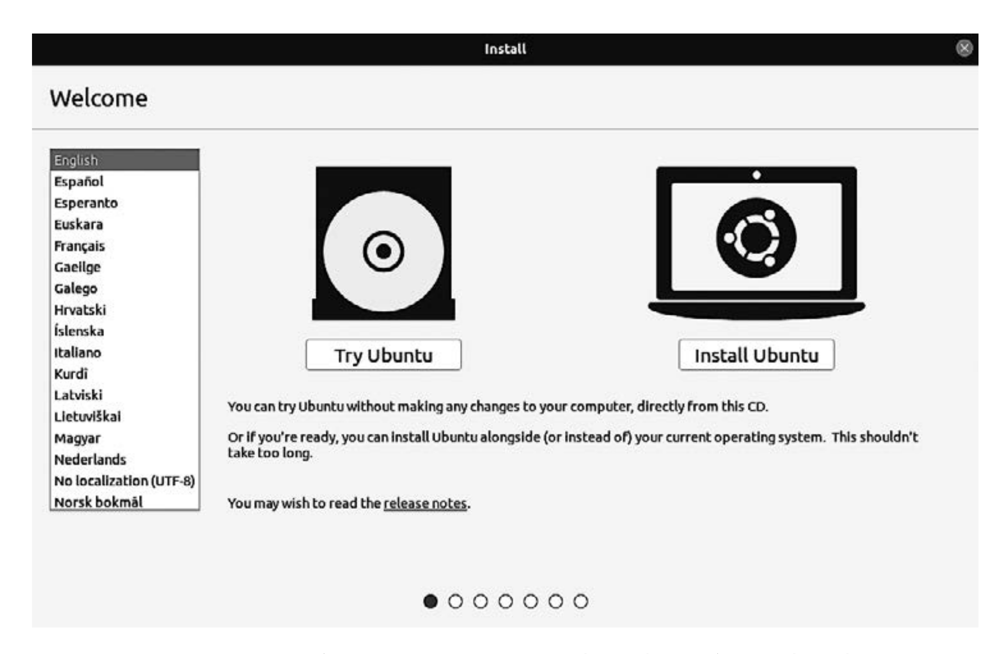


Рис. 1.10. Выбор из двух вариантов: Try Ubuntu (Попробовать Ubuntu) и Install Ubuntu (Установить Ubuntu)

Когда установка будет запущена, программа проведет вас через несколько шагов. Сначала она предложит вам выбрать язык и раскладку клавиатуры.

Затем, если на компьютере имеется беспроводной сетевой интерфейс, предложит настроить его, не дожидаясь завершения установки.

Далее вам будет предложено настроить некоторые параметры установки. На экране Updates and other software (Обновления и другое программное обеспечение) выберите вариант Normal installation (Обычная установка) (рис. 1.11).

На следующем экране выберите Erase disk and install Ubuntu (Очистить диск и установить Ubuntu), а затем нажмите кнопку Install Now (Установить сейчас) (рис. 1.12).

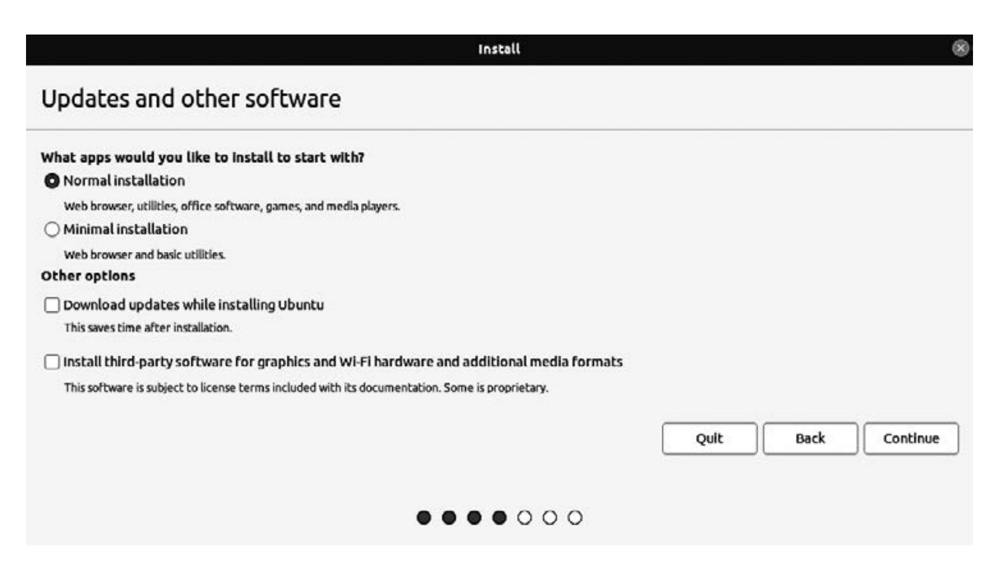


Рис. 1.11. Выбран вариант Normal installation (Обычная установка)

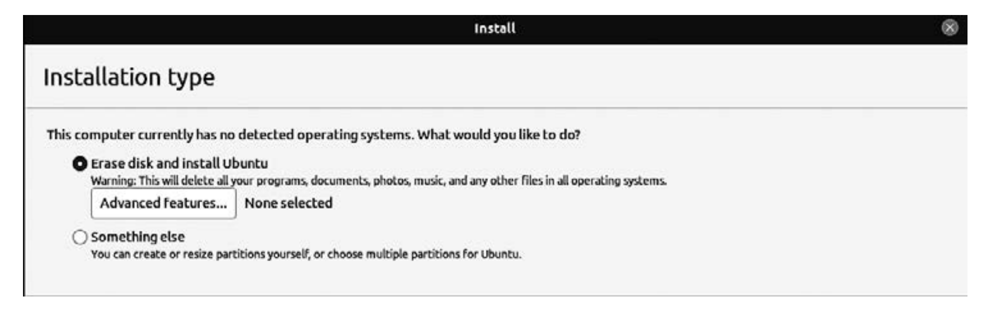


Рис. 1.12. Выбран вариант Erase disk and install Ubuntu (Очистить диск и установить Ubuntu)

На следующем экране в ответ на вопрос Write the changes to disk? (Записать изменения на диск?) нажмите кнопку Continue (Продолжить). Далее будет предложено еще несколько экранов: для настройки часового пояса, создания пользователя, пароля и имени хоста, после чего начнется собственно установка. Вам не требуется что-либо делать до окончания установки. Когда установка завершится, в ответ на предложение перезапустить компьютер извлеките установочный носитель и нажмите клавишу Enter. После перезапуска вам будет предложено задать несколько дополнительных настроек, после чего вы сможете поэкспериментировать со своей свежеустановленной Ubuntu Linux.

Комментарий

Большинство дистрибутивов Linux имеют похожий процесс установки: загрузка с установочного носителя, выбор простой или нестандартной установки. Некоторые дистрибутивы просят ответить на все вопросы (такие как ввод имени пользователя и пароля) перед установкой; другие — после первой перезагрузки.

Программы установки Linux обычно имеют кнопку возврата, давая возможность вернуться назад и изменить настройки. Процедуру установки можно прервать в любой момент; правда, при этом система может остаться в состоянии, непригодном для использования. В этом нет ничего страшного, поскольку установку можно запустить снова и пройти всю процедуру до конца.

Вы можете переустанавливать систему столько раз, сколько захотите, не волнуясь о лицензионных ключах, за исключением корпоративных дистрибутивов, для установки которых требуются регистрационные ключи (Red Hat, SUSE или Ubuntu с платной поддержкой).

Дополнительная информация

- Документация для Ubuntu (https://help.ubuntu.com).

1.8. Настройка дисковых разделов

Задача

Реализовать свою схему разделов жесткого диска.

Решение

В этом рецепте мы вернемся к примеру установки Ubuntu из рецепта 1.7 и реализуем свою схему разделов жесткого диска.



Весь диск будет очищен

В этом рецепте будет создана новая таблица разделов жесткого диска, а все прежнее его содержимое — стерто.

Разбить диск на разделы можно разными способами. В табл. 1.1 показана схема деления, которую я предпочитаю использовать на своих рабочих станциях с Linux.

Таблица 1.1. Пример схемы разделов диска

Имя раздела	Тип файловой системы	Точка монтирования
/dev/sda1	ext4	/boot
/dev/sda2	ext4	/
/dev/sda3	ext4	/home
/dev/sda4	ext4	/tmp
/dev/sda5	ext4	/var
/dev/sda6	swap	

Дойдя до экрана Installation Type (Тип установки), выберите Something else (Другой вариант), чтобы продолжить установку по нестандартному пути (рис. 1.13).

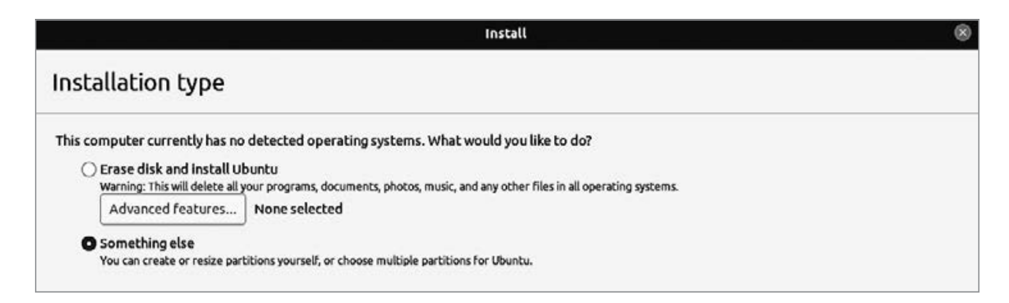


Рис. 1.13. Выбор типа установки

Когда откроется экран со списком разделов на диске, очистите весь диск, нажав кнопку New Partition Table (Новая таблица разделов). После этого откроется экран, изображенный на рис. 1.14.

Для создания нового раздела щелкните на строке free space (свободное место), таким образом выбрав ее, затем нажмите кнопку со значком «плюс», +, чтобы добавить новый раздел. В открывшемся диалоге укажите размер раздела, тип файловой системы и точку монтирования. Например, на рис. 1.15 показано, как заполнить поля диалога, чтобы создать раздел размером 500 Мбайт, который монтируется в каталог `/boot`.

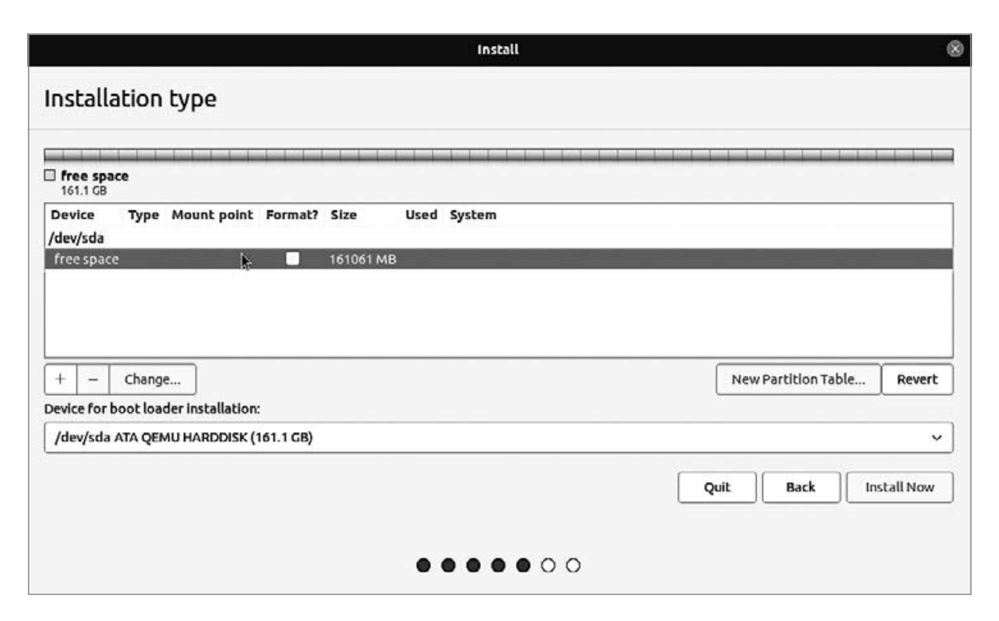


Рис. 1.14. Создание новой таблицы разделов

Create partition
Size: 500 – + MB
Type for the new partition: Primary
Logical
Location for the new partition: Beginning of this space
End of this space
Use as: Ext4 journaling file system
Mount point: /boot
Cancel
OK

Рис. 1.15. Создание загрузочного раздела

Продолжайте щелкать на строке free space (свободное место) и на кнопке со значком +, пока не создадите все разделы. На рис. 1.16 показан конечный результат: здесь созданы разделы для файловых систем `/boot`, `/home`, `/var`, `/tmp`, а также раздел подкачки (`swap`).

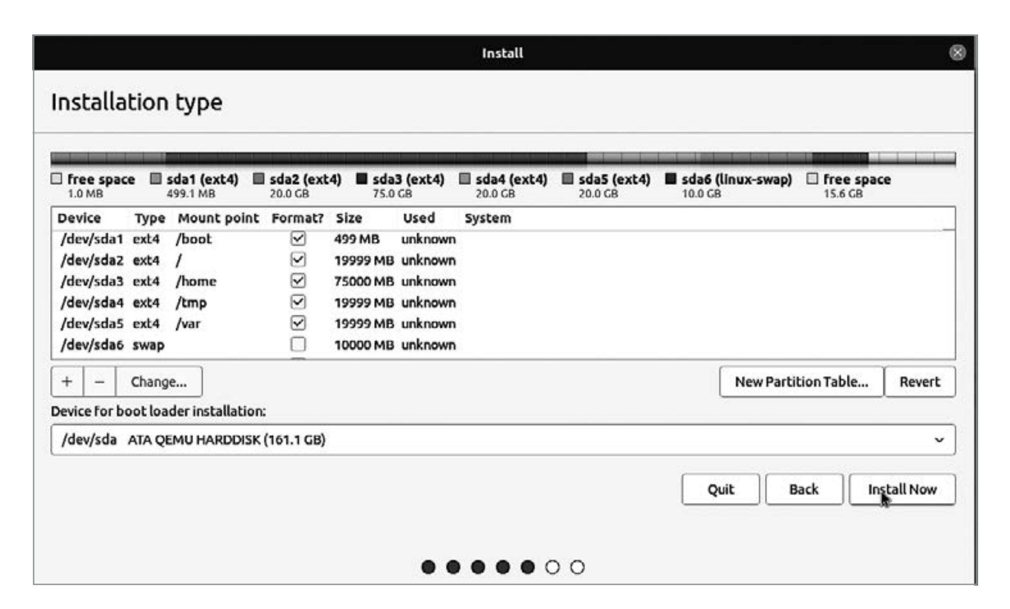


Рис. 1.16. Разделы созданы, и можно продолжать установку



Выбор разделов для форматирования

Обратите внимание на флажки Format? (Форматировать?) в таблице со списком разделов. Все вновь создаваемые разделы следует форматировать в выбранную файловую систему.

Комментарий

В этом рецепте все разделы форматируются в файловой системе Ext4, но вообще можно использовать любые файловые системы, какие захотите; дополнительные подробности см. в главе 11.

Дисковые разделы похожи на группу отдельных физических дисков. Каждый из них независим и может иметь свою файловую систему. Выбираемые файловые системы и их размеры зависят от предназначения системы. Если вам нужно много места для хранения данных, то раздел для каталога `/home` должен быть большим. Это может быть даже отдельный диск.

Создание отдельного раздела для `/boot` упрощает управление мультизагрузочными системами, поскольку отделяет загрузочные файлы от устанавливаемых операционных систем; 500 Мбайт более чем достаточно.

Создание отдельного раздела для корневой файловой системы / упрощает восстановление или замену ядра Linux. Для большинства дистрибутивов 30 Гбайт более чем достаточно, за исключением случаев, когда используется файловая система Btrfs — в этом случае следует выделить 60 Гбайт, чтобы иметь достаточно места для хранения моментальных снимков файловой системы.

Каталог `/home` желательно поместить в отдельный раздел, чтобы изолировать его от корневой файловой системы и иметь возможность переустанавливать Linux, не касаясь этого каталога. Более того, его можно даже поместить на отдельный диск.

Каталоги `/var` и `/tmp` могут заполняться неконтролируемыми процессами. Размещение их в отдельных разделах предотвращает отрицательное влияние вероятных сбоев на другие файловые системы. Я обычно выделяю для них по 20 Гбайт, но на высоконагруженных серверах эти разделы должны быть больше.

Создание раздела подкачки размером, равным объему оперативной памяти, позволяет организовать приостановку с сохранением на диск.

Дополнительная информация

- Подраздел «Комментарий» в рецепте 3.9, посвященный режимам ожидания и сна.
- Глава 8.
- Глава 9.

1.9. Сохранение существующих разделов

Задача

На диске имеется отдельный раздел для каталога `/home`, и его нужно сохранить после переустановки Linux.

Решение

В рецептах 1.7 и 1.8 мы полностью очистили диск, создав новую таблицу разделов. Но если у вас есть разделы, которые вы хотите сохранить, такие как `/home` или любой другой общий каталог, то вместо того, чтобы создавать новую

таблицу разделов, следует отредактировать существующие разделы. При этом можно удалять существующие и создавать новые или повторно использовать имеющиеся разделы.

В следующем примере установки Ubuntu предполагается, что раздел `/dev/sda3` занимает каталог `/home`, который нужно сохранить. Щелкните на нем правой кнопкой мыши и выберите в контекстном меню пункт Change (Изменить). Затем назначьте этому разделу точку монтирования `/home` и убедитесь, что флажок Format? (Форматировать?) **не** установлен (рис. 1.17). Если включить форматирование или изменить тип файловой системы, то все данные в этом разделе будут удалены.

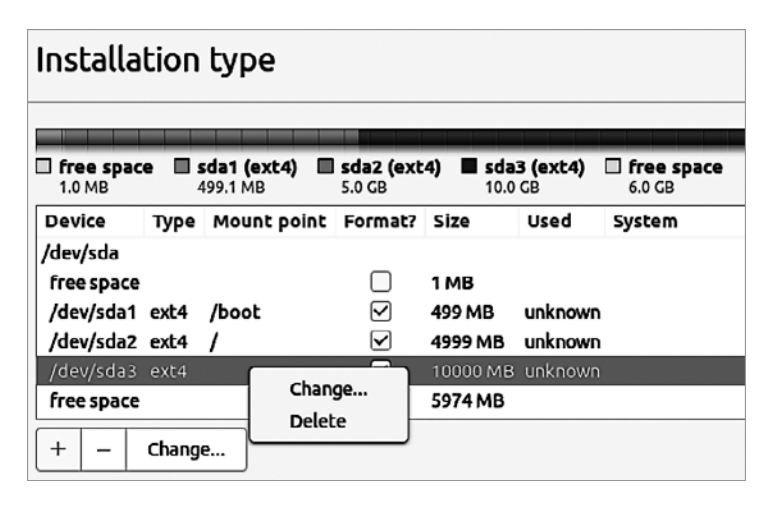


Рис. 1.17. Сохранение раздела /dev/sda3

Комментарий

В рецепте 1.8 вы найдете дополнительные подробности настройки разделов и информацию о том, какие файловые системы желательно разместить в отдельных разделах.

Дополнительная информация

- Глава 8.
- Глава 9.

1.10. Выбор пакетов для установки

Задача

Вам не нравится выбор по умолчанию пакетов для установки и хотелось бы установить программные пакеты по своему выбору. Например, потому, что вам нужна рабочая станция для разработки ПО, веб-сервер, централизованный сервер резервного копирования, рабочая станция для создания видео- или аудиороликов, для издательского дела или просто хотелось бы установить другие офисные приложения.

Решение

В разных дистрибутивах Linux немного по-разному организовано управление пакетами. В этом рецепте вы увидите примеры для openSUSE и Fedora Linux. openSUSE поддерживает несколько способов установки из одного установочного образа, а Fedora Linux имеет несколько разных установочных образов.

Эти два примера типичны для дистрибутивов Linux общего назначения.

Хочу напомнить, что после установки системы вы сможете установить дополнительное или удалять ненужное программное обеспечение.

openSUSE

Программа установки openSUSE поддерживает простую установку по умолчанию и предлагает дополнительные возможности настройки. Она имеет два экрана управления выбором пакетов. Первый (рис. 1.18) предлагает выбрать роль системы, например: рабочий стол с графической средой KDE или GNOME, базовый рабочий стол с оконным менеджером IceWM, сервер без графической среды или сервер транзакций без графической среды. Каждая роль предусматривает установку предопределенного набора пакетов. Вы можете установить одну из них как есть или выбрать пакеты для установки или удаления.

Каждую роль можно дополнительно настроить, как вы увидите несколько позже (рис. 1.19).

Щелкните на ссылке Software (Программное обеспечение), чтобы открыть экран выбора пакетов. На этом экране отображаются *шаблоны* openSUSE — группы взаимосвязанных пакетов, которые можно установить одним щелчком. Лично мне нравится рабочий стол Xfce, поэтому я всегда добавляю его для установки (рис. 1.20).

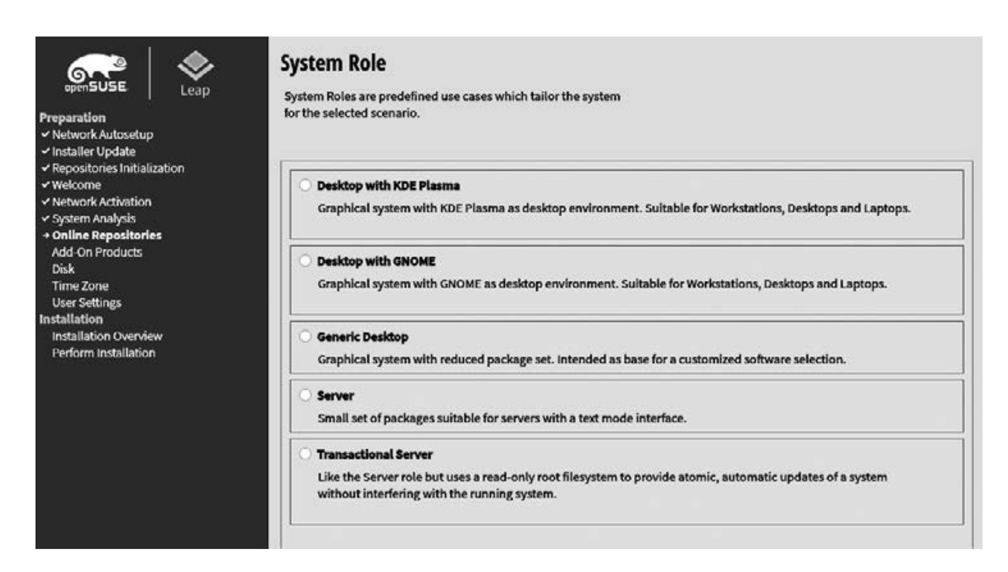


Рис. 1.18. Выбор роли системы в openSUSE

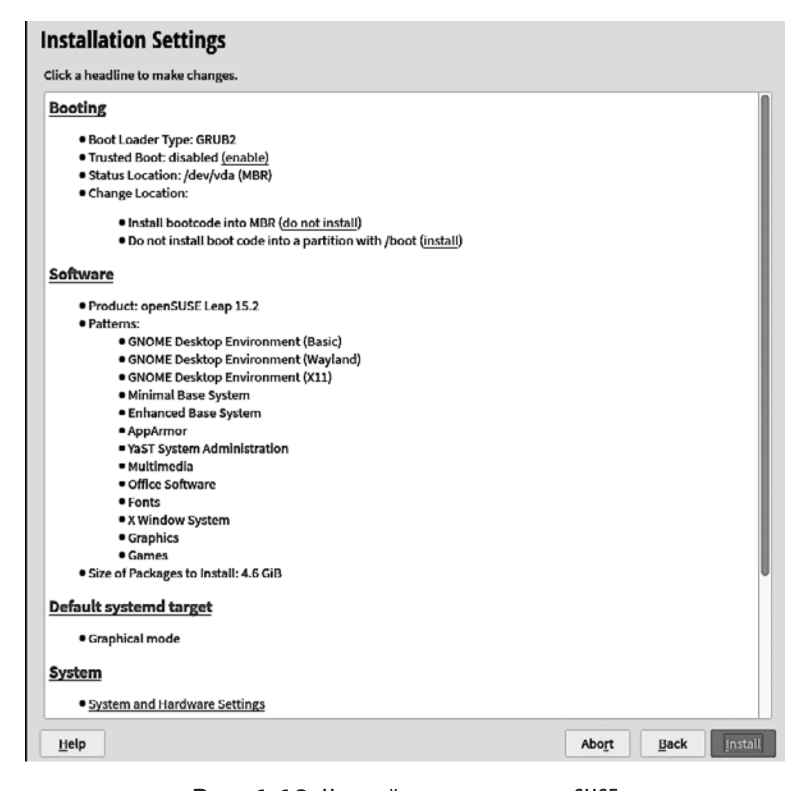


Рис. 1.19. Настройки установки openSUSE

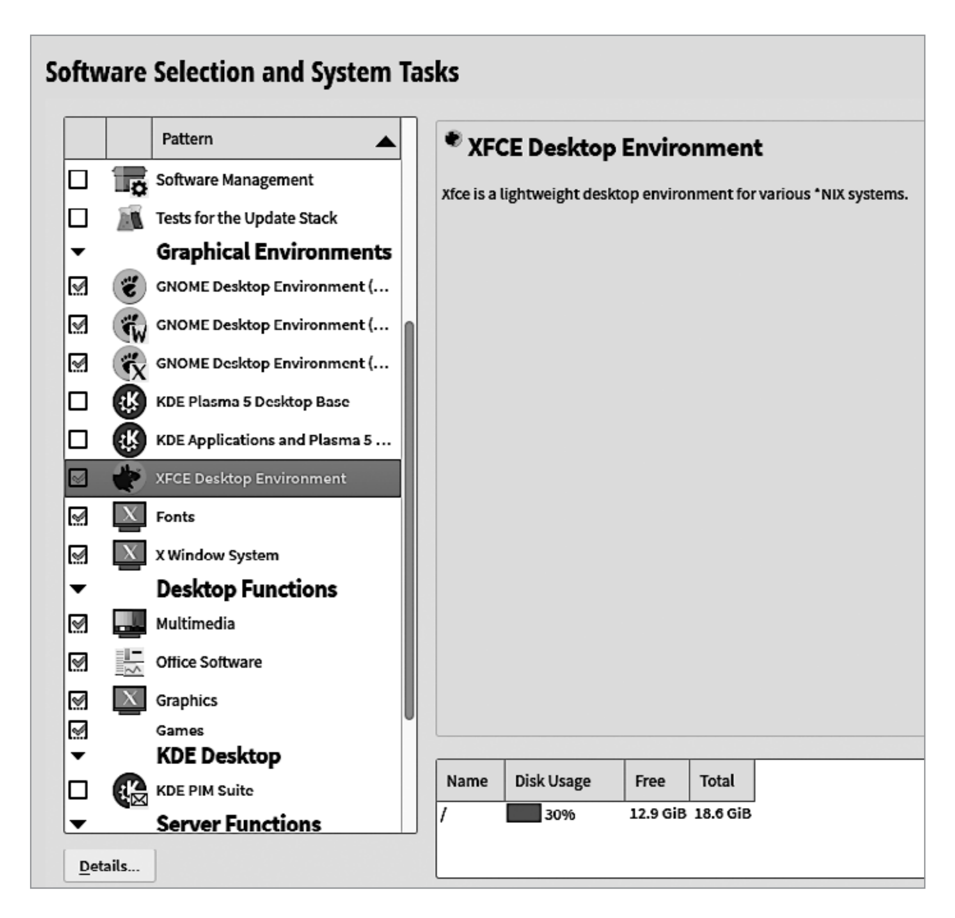


Рис. 1.20. Шаблоны openSUSE с наборами программных пакетов

Обратите внимание на кнопку Details (Подробности) внизу слева. Если нажать ее, то откроется экран с несколькими вкладками, позволяющий выбрать отдельные пакеты (рис. 1.21). На этом экране вы найдете огромное количество информации: списки пакетов для каждого шаблона, группы пакетов, списки репозиториев, сводные данные об установке, зависимости и сведения о каждом пакете. Используйте окно справа, чтобы выбрать или отменить выбор пакетов из каждого шаблона. Программа установки автоматически разрешит зависимости после внесения изменений.

Выбрав пакеты для установки, вы вернетесь обратно на экран Installation Settings (Настройки установки) и получите еще один шанс изменить настройки установ-

ки. Нажмите кнопку Install (Установить), а затем зеленую кнопку Next (Далее), чтобы завершить установку.

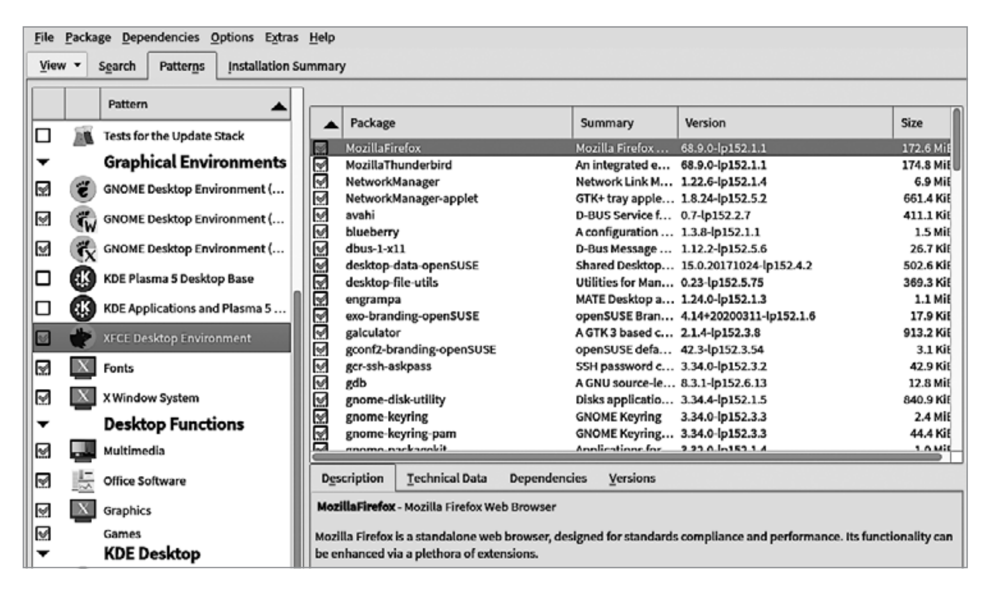


Рис. 1.21. Выбор отдельных пакетов в openSUSE

Fedora Linux

Установочные образы Fedora Linux Workstation и Fedora Linux Server предлагают только возможность настроить разметку диска, но не позволяют выбрать пакеты для установки. Чтобы получить более широкие возможности настройки, нужно скачать образ для установки по сети размером 600 Мбайт из раздела Fedora Alternative Downloads (https://oreil.ly/JW9J8). Он подписан как Fedora Server, но в процессе установки дает возможность выбирать пакеты без всяких ограничений для любого типа установки. Настройте установку по своему желанию на экране Installation Summary (Обзор установки) (рис. 1.22).

Уделите внимание всем параметрам установки на экране Installation Summary (Обзор установки): Software Selection (Выбор программного обеспечения), User Creation (Создание пользователя), Installation Destination (Место установки), Keyboard (Клавиатура), Time & Data (Время и дата), Network & Host Name (Сеть и имя хоста). Щелкните на ссылке Software Selection (Выбор программного обеспечения), чтобы открыть экран выбора пакетов для установки (рис. 1.23).

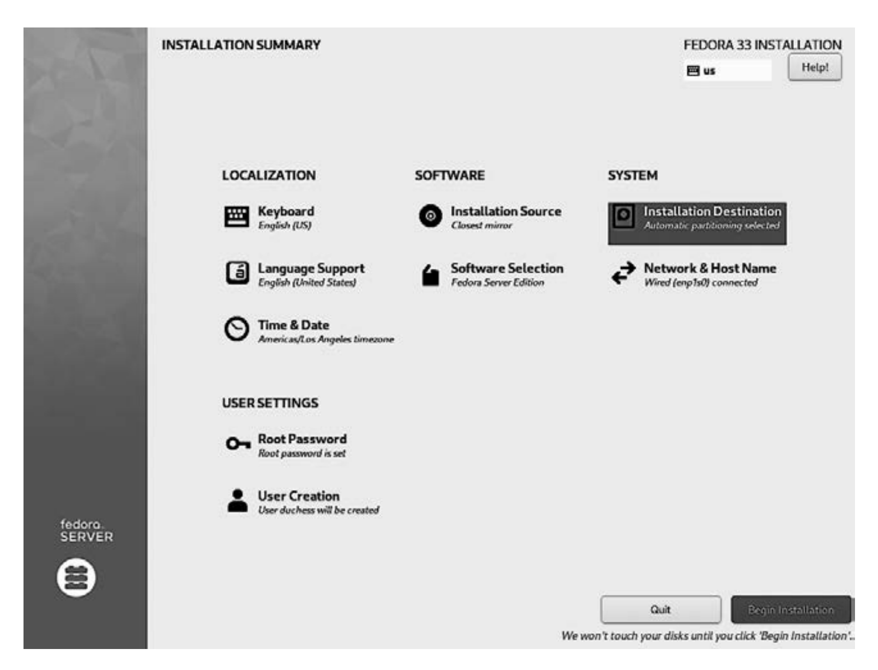


Рис. 1.22. Сетевая установка Fedora Linux

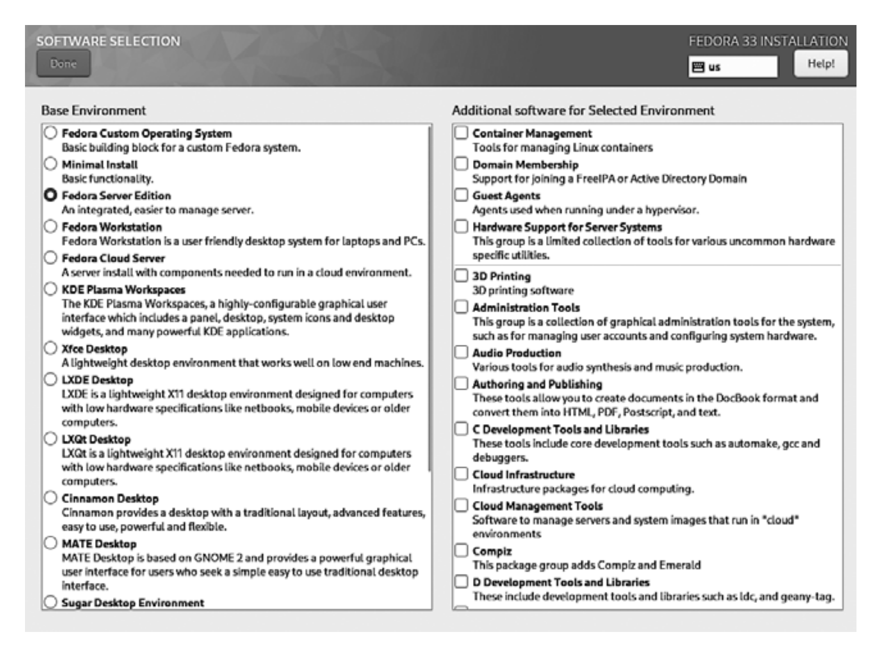


Рис. 1.23. Выбор пакетов для установки в Fedora Linux

Завершив выбор пакетов, нажмите кнопку Done (Готово); после этого вновь откроется экран Installation Summary (Обзор установки). Завершив настройку всех параметров установки, нажмите кнопку Begin Installation (Начать установку), после чего установка продолжится полностью в автоматическом режиме.

Комментарий

Независимо от выбранного дистрибутива Linux, прочтите его документацию и примечания к выпуску. Эти документы содержат важную информацию, избавляющую от многих проблем. Поищите также форумы, списки рассылки и страницы «Википедии», где можно получить помощь.

Вы можете установить столько сред рабочего стола, сколько пожелаете, а затем выбрать ту, которую захотите использовать при входе в систему. Кнопка выбора рабочих столов обычно довольно маленькая и малозаметная; например, на рис. 1.24 показан экран входа в Ubuntu по умолчанию, скрывающий кнопку выбора среды рабочего стола, пока не будет выбрано имя пользователя. Наибольшей популярностью пользуются рабочие столы Xfce, Lxde, GNOME и KDE. GNOME используется по умолчанию в Ubuntu, openSUSE и Fedora.



Рис. 1.24. Выбор графической среды рабочего стола

Дополнительная информация

- Документация для openSUSE (https://oreil.ly/AupNr).
- SUSE Transactional Updates (https://oreil.ly/mTyuV).

1.11. Мультизагрузка нескольких дистрибутивов Linux

Задача

Установить на компьютер несколько дистрибутивов Linux и настроить возможность выбора дистрибутива для запуска во время загрузки.

Решение

Вы можете установить столько дистрибутивов Linux, сколько поместится на вашем жестком диске (или дисках). Если у вас уже должен быть установлен один дистрибутив Linux с отдельным разделом `/boot`, то вам остается выполнить следующие шаги.

1. Выделите достаточно свободного дискового пространства для установки нового дистрибутива Linux на том же жестком диске, где находится существующая система Linux, или на отдельном жестком диске, внутреннем или внешнем.
2. Выпишите на лист бумаги разделы, принадлежащие первой установленной системе Linux, чтобы случайно не перезаписать или не удалить их.
3. Смонтируйте раздел `/boot` в каждой вновь устанавливаемой системе и не форматируйте его.
4. Загрузитесь с установочного носителя, настройте установку новой системы в свободное место на диске.

Программа установки автоматически найдет существующую систему Linux и добавит новую систему в меню загрузки. После завершения установки вы увидите меню загрузки, подобное изображенному на рис. 1.25, в котором присутствуют пункты для выбора Linux Mint и Ubuntu.

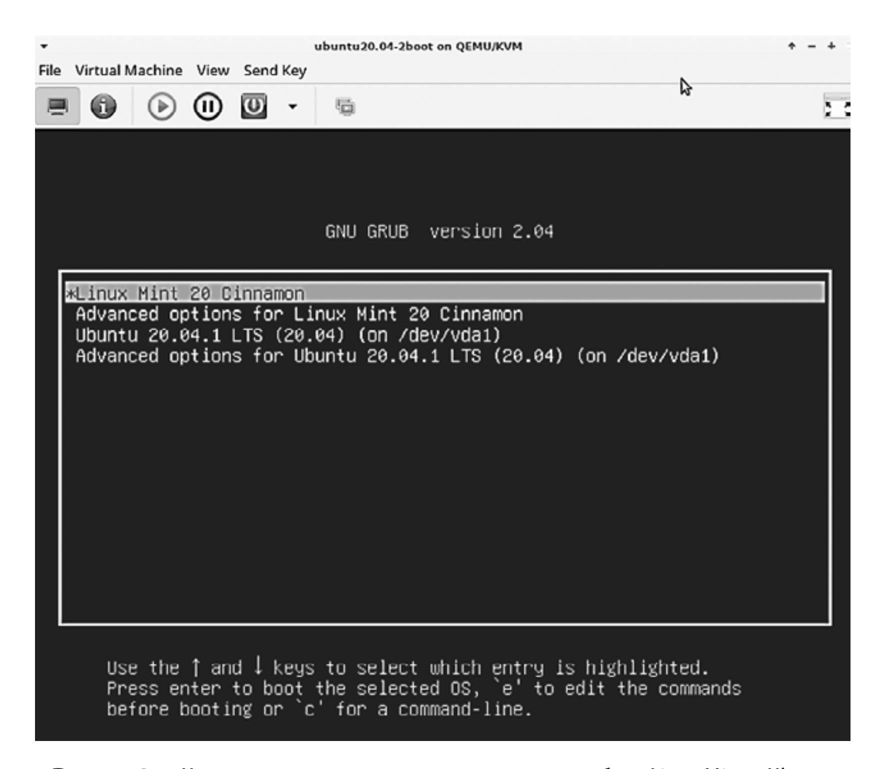


Рис. 1.25. Новое загрузочное меню с пунктами для выбора Linux Mint и Ubuntu

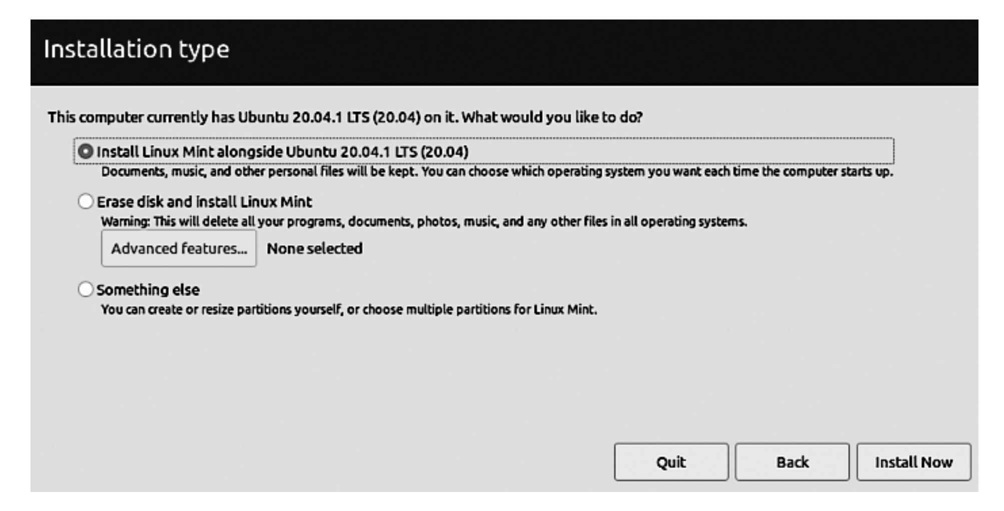


Рис. 1.26. Установка Linux Mint после Ubuntu

Комментарий

Чтобы освободить место на жестком диске, можно сжать существующие разделы (см. рецепт 9.7) перед запуском установки. Для большей безопасности сжимаемые разделы лучше размонтировать. Кроме того, некоторые файловые системы не позволяют сжать смонтированный раздел. Используйте SystemRescue для сжатия разделов (см. рецепт 19.12).

Большинство программ установки Linux достаточно интеллектуальны, чтобы распознавать уже установленные системы Linux и предложить их сохранить. На рис. 1.26 показано окно программы установки Linux Mint, предоставляющей возможность занять весь жесткий диск или установить новую систему рядом с существующей системой Ubuntu без ее удаления.

Дополнительная информация

- Рецепт 8.9.
- Рецепт 9.7.
- Рецепт 19.12.
- Документация с описанием установки выбранного дистрибутива Linux.

1.12. Двухвариантная загрузка с Microsoft Windows

Задача

Установить Linux и Windows на один компьютер.

Решение

Linux и Windows можно установить на один компьютер, а затем выбирать в меню загрузки ту ОС, которую вы захотите использовать.

Первой лучше установить Windows, если она еще не установлена, а затем Linux. Windows любит управлять загрузчиком, поэтому, устанавливая систему Linux второй, вы позволите ей перехватить управление.

Как всегда, перед установкой убедитесь, что у вас есть свежие резервные копии и носитель для восстановления Windows.

После того как Windows будет установлена, запустите установку Linux. Устанавливать Linux можно любым способом: простым или с индивидуальной настройкой разметки диска и набора устанавливаемых пакетов. Однако при установке Linux и Windows на один компьютер важно помнить:

1) если имеется только один жесткий диск, то «устройством для установки загрузчика» должно быть `/dev/sda`;
2) если Windows установлена на одном жестком диске, а Linux устанавливается на второй жесткий диск, то «устройством для установки загрузчика» должен быть диск с Linux. Используйте имя устройства, например `/dev/sdb`, а не раздела, такое как `/dev/sdb1`.

На рис. 1.27 показаны два жестких диска: Windows установлена на `/dev/sda`, а Linux — на `/dev/sdb`.

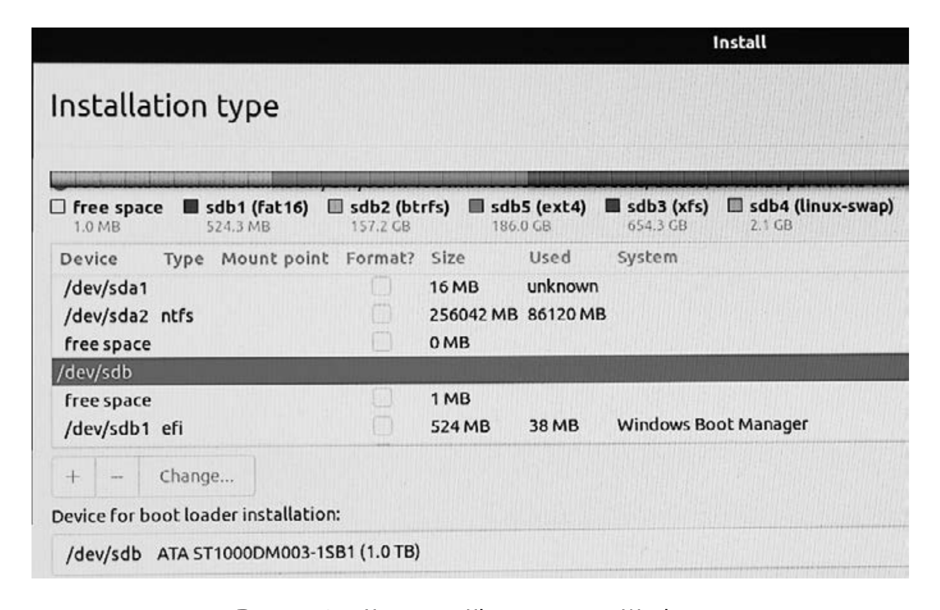


Рис. 1.27. Установка Ubuntu вслед за Windows

Обязательно убедитесь, что устанавливаете Linux в правильное место и не уничтожите Windows. Вы можете разметить диск для системы Linux точно так же,

как если бы устанавливали ее отдельно, и снова будьте предельно осторожны, выбирая разделы для изменения.

Когда разметка диска и конфигурация установки будут настроены, продолжите и завершите установку Linux. После ее завершения и перезагрузки меню GRUB будет содержать пункты для обеих систем (рис. 1.28).



Рис. 1.28. Системы openSUSE и Windows в меню загрузчика GRUB

Комментарий

Вы можете установить на свой компьютер столько систем Linux и Windows, на сколько хватит места на жестких дисках.

Есть и другие способы запустить Linux и Windows на одном компьютере. Windows 10 включает подсистему Windows Subsystem for Linux 2 (WSL 2), которая запускает поддерживаемые дистрибутивы Linux в виртуальной среде. Вы можете запустить Windows в виртуальной машине в Linux, если у вас есть установочный носитель Windows. Виртуальные машины хороши тем, что позволяют запустить несколько операционных систем одновременно, правда, для этого нужны высокопроизводительные процессоры и много памяти.

VirtualBox и QEMU/KVM/Virtual Machine Manager — хорошие бесплатные виртуальные машины, которые работают в Linux.

Дополнительная информация

- Документация для Windows Subsystem for Linux (https://oreil.ly/4cbnk).
- VirtualBox (https://oreil.ly/pI6J6).
- KVM (https://www.linux-kvm.org/page/Main_Page).
- Virtual Machine Manager (https://oreil.ly/5vj6m).
- QEMU (https://oreil.ly/VKBkf).

1.13. Восстановление ключа продукта OEM для Windows 8 или 10

Задача

Вы приобрели компьютер с предустановленной системой Windows 8 или 10, но не можете найти ключ продукта.

Решение

Позвольте Linux сделать это за вас. Выполните следующую команду в системе Linux, установленной на том же компьютере, что и Windows, или загрузившись с диска SystemRescue:

```
$ sudo cat /sys/firmware/acpi/tables/MSDM
MSDMU
DELL CBX3
AMI
FAKEP-RODUC-TKEY1-22222-33333
```

Ключ находится в последней строке.

Если у вас есть возможность загрузиться в Windows, то выполните следующую команду в Windows:

```
C:\Users\Duchess> wmic path softwarelicensingservice get OA3xOriginalProductKey
OA3xOriginalProductKey
FAKEP-RODUC-TKEY1-22222-33333
```

Комментарий

Если у вас нет носителя для восстановления, то вы можете скачать Windows 10 бесплатно. Для установки заново вам понадобится 25-значный ключ продукта OEM.

Дополнительная информация

- Страница для скачивания Windows 10 (https://oreil.ly/rz157).

1.14. Монтирование ISO-образа в Linux

Задача

Имеется скачанный файл `*.iso` с установочным образом Linux, и хотелось бы увидеть, какие файлы и каталоги он содержит. Конечно, можно записать этот образ на DVD или на USB-накопитель, а затем просмотреть его содержимое, однако образ можно открыть, не копируя его на другое устройство.

Решение

В Linux есть псевдоустройство, называемое *петлевым* (loop) устройством. С его помощью можно смонтировать образ `*.iso` так же, как любую другую файловую систему. Выполните следующие действия, чтобы смонтировать образ `*.iso` в петлевое устройство.

Сначала создайте в домашнем каталоге точку монтирования с любым именем. В этом примере я выбрала имя `loopiso`:

```
$ mkdir loopiso
```

Смонтируйте образ `*.iso` в этот новый каталог. В данном примере монтируется установочный образ Fedora Linux:

```
$ sudo mount -o loop Fedora-Workstation-Live-x86_64-34-1.2.iso loopiso
mount: /home/duchess/loopiso: WARNING: device write-protected, mounted read-only
```

После этого откройте смонтированную файловую систему в диспетчере файлов (рис. 1.29).

Вы можете заходить в каталоги, заглядывать в файлы. Но вы не сможете редактировать файлы, поскольку образ смонтирован в режиме «только для чтения».

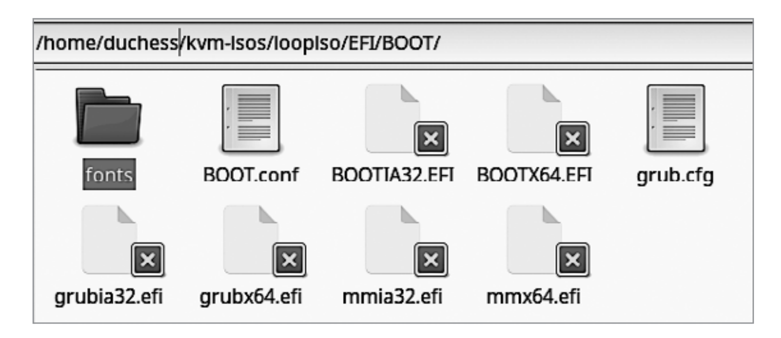


Рис. 1.29. Смонтированный образ Fedora Linux 34

Закончив знакомство с содержимым, размонтируйте образ:

```
$ sudo umount loopiso
```

Комментарий

Петлевое устройство отображает обычный файл в виртуальный раздел и позволяет настроить в этом файле виртуальную файловую систему. Если вы хотите попробовать создать собственный файл образа, то, поискав в Интернете, найдете множество советов. Начните с `man 8 losetup`.

Дополнительная информация

- `man 8 mount`
- `man 8 losetup`

ГЛАВА 2

Управление загрузчиком GRUB

Загрузчик — это программа, загружающая операционную систему после включения компьютера. Чаще других в Linux используется загрузчик GRUB (GRand Unified Bootloader).

GRUB поддерживает ряд полезных возможностей: загрузку нескольких операционных систем на одном ПК, редактирование конфигурации в реальном времени, настраиваемый интерфейс и режимы восстановления. В этой главе мы обсудим их все.



GRUB и GRUB 2

Существует две основные версии GRUB: устаревшая GRUB и более новая GRUB 2. Под названием GRUB 2 подразумеваются версии GRUB 1.99 и выше. Устаревшие версии GRUB — вплоть до 0.97 — прекратили существование в 2005 году. Многие инструкции по GRUB все еще ссылаются на устаревшую версию GRUB и сравнивают ее с GRUB 2. В этой главе я не буду рассказывать о GRUB. Она давно не используется и имеет мало общего с GRUB 2, так что в главе мы сосредоточимся исключительно на GRUB 2.

В одних дистрибутивах Linux используется простое название GRUB без цифрового индекса, в других — название GRUB 2. Например, в Ubuntu есть каталог /boot/grub/ и команда grub-mkconfig, а в Fedora — каталог /boot/grub2/ и команда grub2-mkconfig. Проверьте имена каталогов и команд у себя. В этой главе я воспользуюсь схемой именования, принятой в Ubuntu, за исключением примеров, относящихся к конкретным дистрибутивам.

Запуск компьютера не слишком изменился с тех пор, как в 1940-х годах был создан UNIVAC. Запуск компьютера называется *начальной самозагрузкой (bootstrapping)*, подобно тому как барон Мюнхаузен вытягивал себя из болота за волосы, что в принципе невозможно. Сложность с программируемым компьютером состоит в том, что ему нужны программные инструкции, направляющие его действия, но откуда эти инструкции берутся до загрузки операционной системы?

В современной архитектуре ПК x86_64 начальные инструкции запуска хранятся в микросхеме на материнской плате, а сам процессор устроен так, что при включении начинает выполнять инструкции с адресами, совпадающими с адресами памяти в микросхеме. Можно сказать, что процессор жестко запрограммирован на получение начальных инструкций из определенного адреса. Этот адрес один и тот же для всех компьютеров x86_64, благодаря чему есть возможность комбинировать разные материнские платы и процессоры. (Если вам интересно, то этот адрес называется *вектором сброса (reset vector)*.)

Ниже представлено упрощенное описание того, как происходит загрузка.

Сразу после включения электропитания начинается первый этап. Процессор извлекает инструкции из микропрограммы BIOS/UEFI и инициализирует свои кэши и системную память. После инициализации системной памяти запускается процедура самотестирования при включении (Power On Self-Test, POST), в ходе которой проверяется память и наличие другого оборудования, например клавиатуры, мыши, дисплея и дисков. Возможно, вы замечали, как загораются и гаснут светодиоды на клавиатуре, и слышали шумы внутри корпуса вашего компьютера при проверке дисководов.

Закончив процедуру самотестирования, микропрограмма BIOS/UEFI начинает второй этап загрузки, в ходе которого отыскивает загрузчик на жестком диске и запускает его. Загрузчик GRUB загружает файлы, необходимые для запуска операционной системы, и выводит загрузочное меню.

После вывода загрузочного меню (рис. 2.1) GRUB ждет ввода некоторое время, обычно 5–10 секунд, а затем загружает систему по умолчанию, если вы ничего не сделали за это время. Перемещение по пунктам меню загрузки выполняется с помощью клавиш со стрелками. После нажатия любой клавиши обратный отсчет останавливается, и вы можете не спеша изучить варианты загрузки.

Первый пункт в меню на рис. 2.1 загружает систему. Следующие два открывают подменю с дополнительными параметрами загрузки. После перехода в подменю вы можете вернуться в главное меню, нажав клавишу Esc.

Некоторые дистрибутивы Linux, такие как Fedora и Ubuntu, не отображают загрузочное меню, если установлена только одна операционная система. В этом случае нажмите клавишу Shift сразу после включения электропитания, чтобы увидеть загрузочное меню, которое будет содержать пункт для перехода к настройкам.

Возможно, вы захотите настроить внешний вид и поведение меню GRUB, изменив некоторые параметры в конфигурации GRUB.

Если вы предпочитаете графический инструмент для настройки меню GRUB, то попробуйте GRUB Customizer (рис. 2.2). Этот инструмент доступен в большинстве дистрибутивов Linux как пакет *grub-customizer*, за исключением openSUSE,

в котором имеется модуль управления настройками GRUB (подписанный как Boot Loader (Загрузчик)) в утилите настройки системы YaST.



Рис. 2.1. Загрузочное меню GRUB в openSUSE



Рис. 2.2. GRUB Customizer

2.1. Повторная сборка конфигурационного файла GRUB

Задача

Собрать конфигурацию GRUB после изменения настроек.

Решение

Команды сборки конфигурации GRUB в разных дистрибутивах могут различаться. В Fedora и openSUSE эта команда выглядит так:

```
$ sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

В некоторых дистрибутивах, таких как Ubuntu, она имеет следующий вид:

```
$ sudo grub-mkconfig -o /boot/grub/grub.cfg
```

В Ubuntu Linux имеется также сценарий `update-grub`, который запускает `grub-mkconfig`:

```
$ sudo update-grub
```

Комментарий

Некоторые дистрибутивы Linux предоставляют команды, анализирующие файл с настройками `/etc/default/grub`.

Не забывайте проверять имена файлов и каталогов при редактировании настроек GRUB, поскольку они различаются в разных дистрибутивах Linux.

Дополнительная информация

- Документация для вашей материнской платы, где приводятся сведения о BIOS/UEFI.
- Руководство GNU GRUB Manual (https://oreil.ly/szAiR).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.

2.2. Отображение скрытого меню GRUB

Задача

Ваш любимый дистрибутив Linux скрывает меню GRUB, если на компьютере установлена только одна операционная система, но вам хотелось бы, чтобы оно появлялось перед загрузкой.

Решение

Так поступают некоторые дистрибутивы Linux, включая Ubuntu и Fedora. Вы можете отобразить меню GRUB, нажав и удерживая клавишу Shift после включения электропитания.

Отредактируйте файл `/etc/default/grub`, как показано ниже, чтобы заставить GRUB всегда отображать меню:

```
GRUB_TIMEOUT="10"
GRUB_TIMEOUT_STYLE=menu
```

Если в вашем файле присутствуют строки `GRUB_HIDDEN_TIMEOUT=0` и `GRUB_HIDDEN_TIMEOUT_QUIET=true`, то закомментируйте их, добавив символ решетки (`#`) в начало строки.

После того как файл `/etc/default/grub` будет изменен, соберите конфигурацию GRUB (см. рецепт 2.1).

Комментарий

Параметр `GRUB_HIDDEN_TIMEOUT=0` запрещает отображение меню GRUB, а параметр `GRUB_HIDDEN_TIMEOUT_QUIET=true` запрещает отображение таймера обратного отсчета.

Если установить на компьютер еще одну операционную систему в режиме мультизагрузки, то меню GRUB должно стать видимым автоматически.

Дополнительная информация

- Рецепт 2.1.
- Руководство GNU GRUB Manual (https://oreil.ly/DqiwS).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.

2.3. Загрузка с другим ядром Linux

Задача

Вам интересно узнать смысл дополнительных пунктов в вашем меню GRUB, которые ссылаются на определенные версии ядра Linux, как на рис. 2.3, для чего они нужны и что с ними делать.

```
GNU GRUB  version 2.04

*Ubuntu, with Linux 5.4.0-58-generic
 Ubuntu, with Linux 5.4.0-58-generic (recovery mode)
 Ubuntu, with Linux 5.4.0-42-generic
 Ubuntu, with Linux 5.4.0-42-generic (recovery mode)

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands
   before booting or `c' for a command-line. ESC to return previous
   menu.
```

Рис. 2.3. Пункты в меню GRUB для загрузки разных ядер

Решение

С течением времени система Linux обновляется, но старые ядра Linux сохраняются и добавляются в меню GRUB. Это позволяет загрузить старое и заведомо исправное ядро, если что-то пойдет не так с новым ядром. Старые ядра можно не сохранять и удалять их с помощью диспетчера пакетов.

Комментарий

В прежние времена обновление ядра имело большое значение, поскольку часто это означало исправление ошибок, улучшенную поддержку оборудования, такого как видео-, аудио- и сетевые карты, а также новые возможности, например

поддержку проприетарных форматов файлов и новых протоколов. Изменений было много, и новые ядра нередко работали некорректно, поэтому сохранение возможности загрузки с более старым ядром было обычным делом. В наши дни подобные проблемы встречаются редко, и обновления ядра, как правило, незначительны.

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/qxk2m).
- Архив ядер Linux (https://oreil.ly/l5xyK).

2.4. Устройство конфигурационных файлов GRUB

Задача

Вам известно, что настройка GRUB выполняется немного иначе, чем настройка большинства программ, и хотелось бы узнать, где находятся конфигурационные файлы GRUB и как их использовать для управления GRUB.

Решение

Конфигурационные файлы GRUB находятся в `/boot/grub/`, `/etc/default/grub` и `/etc/grub.d/`. Настройка GRUB — сложная задача, требующая использования множества сценариев и модулей.



GRUB и GRUB 2

Как обсуждалось во вводной части этой главы, в одних дистрибутивах Linux используется простое название GRUB без цифрового индекса, в других — название GRUB2 в именах файлов и командах. В этой главе я использую название GRUB, как принято в Ubuntu, за исключением примеров, относящихся к конкретным дистрибутивам.

Файл `/etc/default/grub` хранит настройки внешнего вида меню GRUB, отображаемого при запуске, которые, например, позволяют скрыть или отобразить меню, применить оформление, добавить фоновое изображение, настроить тайм-аут и задать параметры ядра.

Файлы в `/etc/grub.d/` поддерживают более сложные конфигурации, а `/boot/grub/` хранит файлы изображений и оформлений для настройки внешнего вида меню GRUB.

Главный конфигурационный файл GRUB — `/boot/grub/grub.cfg`. Загрузчик GRUB читает его при запуске. Вы не должны редактировать этот файл, поскольку он создается в процессе сборки конфигурации после внесения изменений в файлы из `/etc/grub.d/` и `/etc/default/grub`.

Конфигурация GRUB автоматически собирается заново после установки любых обновлений, влияющих на процесс загрузки, таких как установка новых и удаление старых ядер.

Комментарий

Если вас интересует создание сценариев, то изучение файлов GRUB послужит отличным практическим примером организации большого количества взаимозависимых сценариев.

Каталог `/etc/grub.d/` содержит целый *комплекс* конфигурационных файлов. Вместо одного гигантского конфигурационного файла настройки GRUB хранятся в нескольких отдельных файлах, и каждый из них отвечает за настройки конкретной задачи. Эти файлы пронумерованы в том порядке, в котором GRUB должен их читать: чем меньше число, тем выше приоритет. Ниже представлен пример из Fedora 32:

```
$ sudo ls -C1 /etc/grub.d/
00_header
01_users
08_fallback_counting
10_linux
10_reset_boot_success
12_menu_auto_hide
20_linux_xen
20_ppc_terminfo
30_os-prober
30_uefi-firmware
40_custom
41_custom
backup
README
```

Каждый из этих файлов является сценарием и должен иметь разрешение на выполнение. Вы можете запретить использовать любой из них, сбросив бит разрешения на выполнение, например, так:

```
$ sudo chmod -x 20_linux_xen
```

Вновь разрешить использовать сценарий можно, установив бит разрешения на выполнение:

```
$ sudo chmod +x 20_linux_xen
```

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/RWh6k).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.
- Глава 6.

2.5. Создание минимального конфигурационного файла GRUB

Задача

Написать минимальную действующую конфигурацию GRUB.

Решение

Ниже представлен простейший файл `/etc/default/grub`, содержащий только самое необходимое для отображения меню и загрузки системы Linux (этот пример предназначен для openSUSE Leap 15.2):

```
# Изменив этот файл, выполните команду 'grub2-mkconfig -o /boot/grub2/grub.cfg'
# для обновления /boot/grub2/grub.cfg

GRUB_DEFAULT=0
GRUB_TIMEOUT=10
GRUB_TIMEOUT_STYLE=menu
```

Помните, как выглядело меню на рис. 2.1? На рис. 2.4 изображено меню для той же системы, но полученное с использованием минимальной конфигурации GRUB.

Есть еще несколько параметров, которые вы можете попробовать добавить: например, разные способы выбора загрузки по умолчанию, фоновое изображение и тема, цвета́ и разрешение экрана. Подробнее об этом рассказывается в подразделе «Комментарий».

```
GNU GRUB  version 2.04

*openSUSE Leap 15.2
 Advanced options for openSUSE Leap 15.2
 Start bootloader from a read-only snapshot

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands
   before booting or `c' for a command-line.
```

Рис. 2.4. Минимальное меню GRUB

Комментарий

В `/etc/default/grub` можно добавить множество других параметров, большинство из которых можно игнорировать. Ниже описаны параметры, которые я считаю наиболее полезными.

- `GRUB_DEFAULT` = определяет пункт в загрузочном меню, выбираемый по умолчанию. Счет пунктов в загрузочном меню начинается с 0, но они не нумеруются явно. Как узнать, какой номер соответствует каждому пункту? К сожалению, нет очевидного способа сделать это, и вам придется вручную пересчитать пункты, содержащие разделы `menuentry`. Типичное определение пункта меню выглядит следующим образом:

  ```
  menuentry 'openSUSE Leap 15.2' --class opensuse --class gnu-linux
    --class gnu --class os
  menuentry_id_option 'gnulinux-simple-102a6fce-8985-4896-a5f9-e5980cb21fdb' {
          load_video
          set gfxpayload=keep
          insmod gzio
          [...]
  ```

  Можно также воспользоваться командой `awk`, как в следующем примере для Ubuntu 20.04:

  ```
  $ sudo awk -F\' '/menuentry / {print i++,$2}' /boot/grub/grub.cfg
  0 Ubuntu
  1 Ubuntu, with Linux 5.8.0-53-generic
  2 Ubuntu, with Linux 5.8.0-53-generic (recovery mode)
  3 Ubuntu, with Linux 5.8.0-50-generic
  4 Ubuntu, with Linux 5.8.0-50-generic (recovery mode)
  5 UEFI Firmware Settings
  ```

Вероятно, нет большого смысла использовать в качестве выбора по умолчанию пункт запуска процедуры восстановления или тестирования памяти, хотя это вполне возможно. Пункт UEFI Firmware Settings (Настройки прошивки UEFI) выполняет вход в настройки BIOS/UEFI вашей системы.

- GRUB_TIMEOUT=10 задает время в секундах, в течение которого меню GRUB будет ждать выбора пользователя перед запуском пункта по умолчанию, а GRUB_TIMEOUT_STYLE=menu отображает меню во время обратного отсчета. Параметр GRUB_TIMEOUT=0 запускает загрузку немедленно, без отображения меню, а GRUB_TIMEOUT=-1 отключает автоматическую загрузку и ждет, пока пользователь сделает выбор.
- GRUB_DEFAULT=saved вместе с GRUB_SAVEDEFAULT=true делает последний выбранный пункт меню пунктом по умолчанию для следующей загрузки.
- GRUB_CMDLINE_LINUX= добавляет параметры ядра Linux для всех пунктов меню.
- GRUB_CMDLINE_LINUX_DEFAULT= определяет параметры ядра только для пункта меню по умолчанию. Типичный пример: GRUB_CMDLINE_LINUX_DEFAULT="quiet splash", который отключает подробный вывод при запуске и отображает графический экран-заставку. На рис. 2.5 показано, как выглядит подробный вывод. Если задан параметр GRUB_CMDLINE_LINUX_DEFAULT="quiet splash", то увидеть подробный вывод можно без изменения настроек, нажав клавишу Esc во время запуска системы.
- GRUB_TERMINAL=gfxterm устанавливает графический режим для отображения экрана GRUB, который поддерживает цвета и графические изображения. Параметр GRUB_TERMINAL=console отключает графический режим.
- GRUB_GFXMODE= устанавливает разрешение экрана в графическом режиме, например: GRUB_GFXMODE=1024x768. Запустите в командной строке GRUB команду set pager=1, а затем videoinfo, чтобы увидеть поддерживаемые режимы (рис. 2.6). Команда set pager=1 позволяет использовать клавиши со стрелками для пролистывания выводимых командами результатов вверх и вниз. Параметр GRUB_GFXMODE=auto выбирает наиболее разумное значение по умолчанию.
- GRUB_BACKGROUND= устанавливает указанное фоновое изображение для меню GRUB (см. рецепт 2.6).
- GRUB_THEME= устанавливает указанную тему оформления меню GRUB (см. рецепт 2.8).

```
[  OK  ] Reached target Local Encrypted Volumes.
[  OK  ] Listening on Syslog Socket.
         Starting Journal Service...
[  OK  ] Mounted Kernel Debug File System.
[  OK  ] Mounted POSIX Message Queue File System.
[  OK  ] Mounted Huge Pages File System.
[  OK  ] Started Load Kernel Modules.
[  OK  ] Started Create list of required static device nodes for the current kernel.
[  OK  ] Started Remount Root and Kernel File Systems.
         Starting udev Coldplug all Devices...
         Starting Create Static Device Nodes in /dev...
         Starting Apply Kernel Variables...
[  OK  ] Started Journal Service.
[  OK  ] Started Create Static Device Nodes in /dev.
[  OK  ] Started Apply Kernel Variables.
[  OK  ] Stopped Entropy Daemon based on the HAVEGE algorithm.
[  OK  ] Started Entropy Daemon based on the HAVEGE algorithm.
         Starting udev Kernel Device Manager...
[  OK  ] Started udev Coldplug all Devices.
[  OK  ] Started Setup Virtual Console.
[  OK  ] Started udev Kernel Device Manager.
[    3.630490] pcieport 0000:00:02.6: pciehp: Failed to check link status
[    3.662015] input: Power Button as /devices/LNXSYSTM:00/LNXPWRBN:00/input/input4
[    3.667655] ACPI: Power Button [PWRF]
[  OK  ] Created slice system-qemu\x2dga.slice.
         Starting Setup Virtual Console...
```

Рис. 2.5. Подробный вывод с информацией о ходе запуска

```
  No info available
grub> videoinfo
List of supported video modes:
Legend: mask/position=red/green/blue/reserved
Adapter `Cirrus CLGD 5446 PCI Video Driver':
  No info available
Adapter `Bochs PCI Video Driver':
  No info available
Adapter `VESA BIOS Extension Video Driver':
  VBE info:   version: 3.0  OEM software rev: 0.0
              total memory: 16384 KiB
  0x100  640 x  400 x  8 ( 640)  Paletted
  0x101  640 x  480 x  8 ( 640)  Paletted
  0x102  800 x  600 x  4 (   0)  Paletted Planar
  0x103  800 x  600 x  8 ( 800)  Paletted
  0x104 1024 x  768 x  4 (   0)  Paletted Planar
  0x105 1024 x  768 x  8 (1024)  Paletted
  0x106 1280 x 1024 x  4 (   0)  Paletted Planar
  0x107 1280 x 1024 x  8 (1280)  Paletted
  0x10d  320 x  200 x 15 ( 640)  Direct color, mask: 5/5/5/1  pos: 10/5/0/15
  0x10e  320 x  200 x 16 ( 640)  Direct color, mask: 5/6/5/0  pos: 11/5/0/0
  0x10f  320 x  200 x 24 ( 960)  Direct color, mask: 8/8/8/0  pos: 16/8/0/0
  0x110  640 x  480 x 15 (1280)  Direct color, mask: 5/5/5/1  pos: 10/5/0/15
  0x111  640 x  480 x 16 (1280)  Direct color, mask: 5/6/5/0  pos: 11/5/0/0
  0x112  640 x  480 x 24 (1920)  Direct color, mask: 8/8/8/0  pos: 16/8/0/0
  0x113  800 x  600 x 15 (1600)  Direct color, mask: 5/5/5/1  pos: 10/5/0/15
  0x114  800 x  600 x 16 (1600)  Direct color, mask: 5/6/5/0  pos: 11/5/0/0
--MORE--
```

Рис. 2.6. Поддерживаемые видеорежимы

Дополнительная информация

- Рецепт 2.6.
- Рецепт 2.8.
- Руководство GNU GRUB Manual (https://oreil.ly/zIbDg).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.

2.6. Настройка фонового изображения для меню GRUB

Задача

Вас волнует внешний вид вашего меню GRUB, и вам хотелось бы улучшить его.

Решение

В роли фонового изображения можно использовать изображение в формате PNG, восьмибитном JPG или TFA. Изображение может быть любого размера — GRUB автоматически масштабирует его по размеру экрана. В следующем примере я украшу меню GRUB фотографией моей кошки Герцогини, разгуливающей по книжным полкам.

Скопируйте изображение в `/boot/grub/` и добавьте полный путь к файлу изображения в файл `/etc/default/grub`. В моем случае фотография Герцогини находится в файле `/boot/grub/duchess-books.jpg`:

```
GRUB_BACKGROUND="/boot/grub/duchess-books.jpg"
```

Если в файле имеется строка `GRUB_THEME=`, то закомментируйте ее, затем пересоберите конфигурацию GRUB (см. рецепт 2.1).

Вы должны увидеть сообщение Found background: /boot/grub/duchess-books.jpg в выводе команды сборки. Если это сообщение не появилось, значит, где-то в вашей конфигурации есть ошибка.

После того как конфигурация будет пересобрана успешно, перезагрузитесь и получите удовольствие от нового фона в меню GRUB (рис. 2.7).

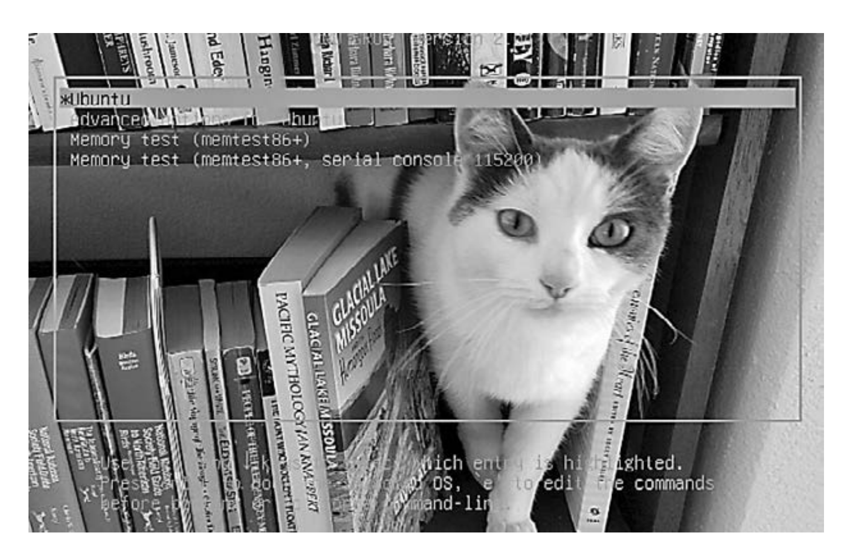


Рис. 2.7. Герцогиня — образованная кошка — красуется в меню GRUB

Шрифты в данном примере плохо читаются, поэтому переходите к рецепту 2.7, чтобы узнать, как изменить их цвет.

Комментарий

Вы можете использовать любое изображение, и оно не обязательно должно находиться в `/boot/grub/`. Однако, помещая файлы изображений в `/boot/grub/`, вы сохраните все ваши настройки GRUB в одном месте и сделаете их доступными для всех систем Linux, установленных в режиме мультизагрузки.

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/zIbDg).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.

2.7. Изменение цвета шрифтов в меню GRUB

Задача

Новый фон смотрится замечательно (см. рис. 2.7), но надписи почти не читаются, и хотелось бы изменить цвет шрифта в меню GRUB.

Решение

Это увлекательное задание, поскольку вы можете быстро просмотреть цвета в командной строке GRUB. А затем отредактировать `/etc/default/grub` и создать новый файл в `/etc/grub.d/` для загрузки ваших цветов, затем пересобрать `/boot/grub/grub.cfg`. Перезагрузитесь, чтобы насладиться фоновым изображением с красивыми цветными шрифтами.

Запустите компьютер и, когда появится меню GRUB, нажмите C, чтобы открыть командную строку GRUB (рис. 2.8).

```
                    GNU GRUB  version 2.04

   Minimal BASH-like line editing is supported. For the first word,
   TAB lists possible command completions. Anywhere else TAB lists
   possible device or file completions. ESC at any time exits.


grub> _
```

Рис. 2.8. Командная строка GRUB

Следующие две команды устанавливают цвета, как показано на рис. 2.9:

```
grub> menu_color_highlight=cyan/blue
grub> menu_color_normal=yellow/black
```

Вы можете поочередно выбрать и проверить каждую пару цветов. Для этого в меню GRUB нажмите клавишу C, чтобы открыть командную строку. Введите свою команду (извините, но копирование и вставка здесь недоступны), нажмите Enter, затем Esc, чтобы вернуться в меню и посмотреть, как оно выглядит. Вы можете перелистывать введенные прежде команды с помощью клавиш со стрелками вверх и вниз, а также редактировать и повторно использовать их, не набирая все заново.

```
GNU GRUB  version 2.04

*openSUSE Leap 15.2
 Advanced options for openSUSE Leap 15.2
 Start bootloader from a read-only snapshot

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands
   before booting or `c' for a command-line.
```

Рис. 2.9. Настройка цветов в меню из командной строки GRUB

Цвета должны указываться в таком порядке: цвет шрифта/цвет фона. Все цвета сплошные, без прозрачности, за одним исключением: когда для фона выбирается черный цвет, он становится прозрачным. Вот почему в `menu_color_normal=` должен указываться черный цвет фона при использовании фонового изображения. Если указать любой другой цвет фона, то изображение будет покрыто цветом фона. Цвет фона `menu_color_highlight=` применяется только к строке, выбранной в данный момент.

Определившись с выбором цветов, сделайте их постоянными. Загрузитесь и создайте новый сценарий в `/etc/grub.d/`. В следующем примере он называется `07_font_colors`. Скопируйте этот текст в точности:

```
#!/bin/sh

      if [ "x${GRUB_BACKGROUND}" != "x" ] ; then
              if [ "x${GRUB_COLOR_NORMAL}" != "x" ] ; then
              echo "set menu_color_normal=${GRUB_COLOR_NORMAL}"
              fi

              if [ "x${GRUB_COLOR_HIGHLIGHT}" != "x" ] ; then
              echo "set menu_color_highlight=${GRUB_COLOR_HIGHLIGHT}"
              fi
      fi
```

Затем дайте разрешение на выполнение:

```
$ sudo chown +x 07_font_colors
```

Потом добавьте следующие строки в файл `/etc/default/grub`, определяющие выбранные цвета:

```
export GRUB_COLOR_NORMAL="yellow/black"
export GRUB_COLOR_HIGHLIGHT="cyan/blue"
```

Пересоберите конфигурацию GRUB (см. рецепт 2.1) и перезагрузите компьютер, чтобы увидеть получившийся результат.

Комментарий

Дополнительную информацию о файлах в `/etc/grub.d/` и о том, почему имена файлов должны начинаться с цифр, вы найдете в рецепте 2.4. По своему опыту могу сказать, что неважно, каким по счету будет запускаться сценарий, определяющий цвета шрифтов, но если вдруг так случится, что его добавление не дает желаемого результата, то попробуйте изменить приоритет. Убедитесь, что файл сценария имеет разрешение на выполнение. Сценарий должен определять следующие параметры:

- `menu_color_highlight` — управляет цветами выделенного пункта в меню;
- `menu_color_normal` — управляет цветами невыделенных пунктов в меню.

Используйте имена цветов в точности так, как они указаны в табл. 2.1, со всеми буквами в нижнем регистре.

Таблица 2.1. Цвета, поддерживаемые GRUB

Поддерживаемые цвета			
black	dark-gray	light-green	magenta
blue	green	light-gray	red
brown	light-cyan	light-magenta	white
cyan	light-blue	light-red	yellow

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/BZHWt).
- Страницы справочного руководства man для GRUB; выполните команду `man -k grub`, чтобы получить полный список страниц.
- `info grub` или `info grub2`.
- Рецепт 2.4.

2.8. Применение темы оформления к меню GRUB

Задача

Вам нравится, когда меню GRUB выглядит красиво, и вы хотите узнать, поддерживает ли GRUB темы оформления для меню и как их установить.

Решение

Считайте, что вам повезло, поскольку для GRUB есть множество красивых тем оформления. Для начала попробуйте отыскать готовые темы с помощью диспетчера пакетов и команды `grep`. Ниже представлен пример для Ubuntu Linux:

```
$ apt search theme | grep grub
```

Здесь `theme | grep grub` используется для фильтрации результатов поиска пакетов. Команда должна вывести список названий пакетов, таких как *grub-theme-breeze*, *grub2-themes-ubuntu-mate* и *grub-breeze-theme*. Установка тем выполняется точно так же, как установка любых других пакетов.

Комментарий

Новая тема должна установиться в `/boot/grub/themes`. Найдите вновь установленную тему, например `/boot/grub/themes/ubuntu-mate`, и файл `theme.txt`. Введите полный путь в `/etc/default/grub`, как показано ниже:

```
GRUB_THEME=/boot/grub/themes/ubuntu-mate/theme.txt
```

Обязательно закомментируйте любые другие строки в конфигурации, относящиеся к оформлению меню, такие как `GRUB_BACKGROUND=`, любые определения цветов шрифта и фона и любые другие темы. Затем пересоберите конфигурацию GRUB (см. рецепт 2.1). В выводе команды вы должны увидеть строку, такую как `Found theme: /boot/grub/themes/ubuntu-mate/theme.txt`.

Если все прошло успешно, то перезагрузитесь, и вы увидите экран, подобный изображенному на рис. 2.10. Если меню отображается неправильно, то проверьте еще раз свою конфигурацию и вывод команды пересборки.

Дополнительная информация

- Темы для GNOME (https://oreil.ly/oLJtx).
- Темы для KDE (https://oreil.ly/SLdkp).
- Руководство GNU GRUB Manual (https://oreil.ly/LeIHu).



Рис. 2.10. Тема Ubuntu MATE для GRUB

2.9. Восстановление незагружающейся системы из приглашения grub>

Задача

Во время запуска система останавливается на приглашении GRUB `grub>` и не загружается. Вы должны знать, как загрузить систему и восстановить работоспособную конфигурацию.

Решение

Когда процесс загрузки останавливается на приглашении `grub>` (рис. 2.11), это означает, что загрузчик нашел каталог `/boot/grub/`, но не может найти корневую файловую систему.

Вам нужно найти корневую файловую систему, ядро Linux и соответствующий файл `initrd`. В командной оболочке GRUB вам доступна вся файловая система.

```
GNU GRUB  version 2.04

Minimal BASH-like line editing is supported. For the first word, TAB lists possible command completions. Anywhere
else TAB lists possible device or file completions. ESC at any time exits.

grub> _
```

Рис. 2.11. Командная оболочка GRUB

Первым делом нужно включить возможность перелистывания страниц, чтобы можно было просматривать длинный вывод, прокручивая его вверх и вниз:

```
grub> set pager=1
```

Выведите список доступных дисков и разделов. GRUB поддерживает свой способ идентификации жестких дисков и разделов. Он нумерует диски, начиная с 0, а разделы — с 1, и маркирует все жесткие диски как `hd`. В работающей системе Linux жесткие диски обозначаются как `/dev/sda`, `/dev/sdb` и т. д.

Команда в следующем примере обнаружила два жестких диска, `hd0` и `hd1`, которые соответствуют устройствам `/dev/sda` и `/dev/sdb`. Раздел `hd0,gpt5` соответствует разделу `/dev/sda5`, а `hd1,msdos1` — разделу `/dev/sdb1`:

```
grub> ls
(hd,0) (hd0,gpt5) (hd0,gpt4) (hd0,gpt3) (hd0,gpt2) (hd0,gpt1)
(hd1) (hd1,msdos1)
```

Как показывает этот вывод, диск `hd0` имеет таблицу разделов `gpt`, а диск `hd1` — старомодную таблицу разделов `msdos`. При использовании номеров разделов в командах метки `gpt` и `msdos` можно опустить.

GRUB может сообщить вам типы файловых систем, универсальные уникальные идентификаторы (Universally Unique IDentifier, UUID) и другую информацию о разделах:

```
grub> ls (hd0,3)
  Partition hd0,3: filesystem type ext* - Last modification time 2021-12-29
  01:17:58 Tuesday, UUID 5c44d8b2-e34a-4464-8fa8-222363cd1aff - Partition start
  at 526336KiB -
Total size 20444160KiB
```

Вам нужно найти каталог `/boot`. Предположим, вы помните, что он находится в корневой файловой системе во втором разделе; начните поиск с него. Косая черта после имени раздела означает «список всех файлов и каталогов в разделе»:

```
grub> ls (hd0,2)/
bin   dev  home  lib64    media  opt   root  sbin  sys  usr
boot  etc  lib   lost+found   mnt    proc  run   srv   tmp  var
```

Все загрузочные файлы находятся в каталоге `/boot`:

```
grub> ls (hd0,2)/boot
efi/ grub/ System.map-5.3.18-lp152.57-default config-5.3.18-lp152.57-default
initrd-5.3.18-lp152.57-default vmlinuz vmlinuz-5.3.18-lp152.57-default
sysctl.conf-5.3.18-lp152.57-default vmlinux-5.3.18-lp152.57-default.gz
```

Все, что нужно для загрузки системы, находится здесь. Установите раздел корневой файловой системы, ядро и образ `initrd`:

```
grub> set root=(hd0,2)
grub> linux /boot/vmlinuz-5.3.18-lp152.57-default root=/dev/sda2
grub> initrd /boot/initrd-5.3.18-lp152.57-default
grub> boot
```



Автодополнение клавишей Tab

Подобно Bash, командная оболочка GRUB поддерживает автодополнение команд клавишей Tab. Это означает, что вы можете, например, начать вводить /boot/vml, затем нажать клавишу Tab, чтобы дополнить строку до конца или получить список возможных вариантов.

Если имеется несколько файлов `vmlinuz` и `initrd`, то используйте наиболее новые с совпадающими номерами версий. Если все команды верны, то система загрузится и вы сможете исправить конфигурацию GRUB (см. рецепт 2.11).

Комментарий

Когда каталог `/boot` находится в отдельном разделе, вы не увидите никаких других каталогов, поскольку он находится не в корневой файловой системе.

`vmlinuz-5.3.18-lp152.57-default` — сжатое ядро Linux.

`initrd-5.3.18-lp152.57-default` — RAM-диск с временной корневой файловой системой, используемой только в начале запуска системы.

Сбой загрузки может быть вызван повреждением файлов; добавлением, удалением или перемещением жестких дисков; установкой или удалением операционных систем; или повторной разметкой диска. Если вы не можете выйти в командную строку GRUB, то обращайтесь к главе 19, чтобы узнать, как спасти вашу систему с помощью SystemRescue.

Вы можете попрактиковаться в использовании командной оболочки `grub>`, нажав клавишу `C` в загрузочном меню GRUB. Это достаточно безопасно, поскольку внесенные изменения не сохраняются после перезагрузки.

Дополнительная информация

- Руководство GNU GRUB (https://oreil.ly/8SdwS).

2.10. Восстановление незагружающейся системы из приглашения grub rescue>

Задача

Во время запуска система останавливается на приглашении GRUB `grub rescue>` и не загружается. Вы должны знать, как загрузить систему и восстановить работоспособную конфигурацию.

Решение

Если процесс загрузки останавливается на приглашении `grub rescue>` (рис. 2.12) аварийной командной оболочки, это означает, что загрузчик не нашел каталог `/boot`. Но не спешите волноваться — вы сможете найти его в командной строке GRUB, загрузить систему, а затем исправить конфигурацию.

```
Booting from Hard Disk...
error: file `/boot/grub/i386-pc/normal.mod' not found.
Entering rescue mode...
grub rescue>
```

Рис. 2.12. Аварийная командная оболочка GRUB

Выведите список разделов:

```
grub rescue> ls
(hd0) (hd0,gpt5) (hd0,gpt4) (hd0,gpt3) (hd0,gpt2) (hd0,gpt1)
(hd1) (hd1, msdos1)
```

Эта командная оболочка не поддерживает автодополнение клавишей Tab и возможность прокрутки длинного вывода, поэтому вам придется вводить все команды вручную, от начала до конца.

GRUB может сообщить вам типы файловых систем, UUID и другую информацию о разделах:

```
grub rescue> ls (hd0,3)
    Partition hd0,3: filesystem type ext* - Last modification time 2021-12-29
    01:17:58
Tuesday, UUID 5c44d8b2-e34a-4464-8fa8-222363cd1aff - Partition start at
526336KiB -
Total size 20444160KiB
```

Если вы не знаете, в каком разделе находится каталог `/boot`, то вам придется просмотреть все разделы. При использовании номеров разделов в командах метки `gpt` и `msdos` можно опустить. Косая черта после имени раздела означает «список всех файлов и каталогов в разделе»:

```
grub rescue> ls (hd0,2)/
bin   dev  home  lib64    media  opt   root  sbin  sys  usr
boot  etc  lib   lost+found  mnt    proc  run   srv   tmp  var
```

Ура! Каталог `/boot` нашелся в корневой файловой системе. Выведите список файлов в `/boot`:

```
grub rescue> ls (hd0,2)/boot
efi/ grub/ System.map-5.3.18-lp152.57-default config-5.3.18-lp152.57-default
initrd-5.3.18-lp152.57-default vmlinuz vmlinuz-5.3.18-lp152.57-default
sysctl.conf-5.3.18-lp152.57-default vmlinux-5.3.18-lp152.57-default.gz
```

В приглашении `grub rescue>` необходимо выполнить несколько дополнительных команд: вы должны указать, где находится каталог `/boot/grub`, а затем загрузить модули ядра `normal` и `linux`, которые находятся в `/boot/grub/i386-pc` (вместе со многими другими модулями ядра, которые GRUB использует при запуске). Модуль `normal` изменяет режим загрузки с аварийного на нормальный, а `linux` запускает системный загрузчик:

```
grub rescue> set prefix=(hd0,2)/boot/grub
grub rescue> set root=(hd0,2)
grub rescue> insmod normal
grub rescue> insmod linux
```

Когда модули `normal` и `linux` будут загружены, включится поддержка автодополнения клавишей Tab. Кроме того, можно включить прокрутку вывода командой `set pager=1`, чтобы получить возможность использовать клавиши со стрелками для перехода к предыдущим командам. После этого нужно сообщить загрузчику GRUB, где тот сможет найти ядро и файл `initrd`:

```
grub> linux /boot/vmlinuz-5.3.18-lp152.57-default root=/dev/sda2
grub> initrd /boot/initrd-5.3.18-lp152.57-default
grub> boot
```

Если имеется несколько файлов `vmlinuz` и `initrd`, то используйте наиболее новые с совпадающими номерами версий. Если все команды верны, то система загрузится и вы сможете исправить конфигурацию GRUB (см. рецепт 2.11).

Комментарий

Когда каталог `/boot` находится в отдельном разделе, вы не увидите никаких других каталогов, поскольку он находится не в корневой файловой системе.

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/6REHG).

2.11. Переустановка конфигурации GRUB

Задача

Вы смогли загрузить свою систему из командной строки GRUB и теперь хотели бы исправить конфигурацию.

Решение

Внимательно проверьте конфигурационные файлы GRUB на наличие ошибок. Внеся исправления, пересоберите конфигурацию GRUB (см. рецепт 2.1). После этого нужно переустановить GRUB. В следующем примере переустановка производится в `/dev/sda`:

```
$ sudo grub-mkconfig -o /boot/grub/grub.cfg
$ sudo grub-install /dev/sda
```



Используйте корректную команду пересборки

Как отмечалось в рецепте 2.1 и во врезке, посвященной GRUB и GRUB 2, в начале этой главы, проверьте имена каталогов, чтобы избежать ошибок в командах пересборки.

Убедитесь, что верно указали диск, если у вас их несколько, и используйте только имя устройства (например, `/dev/sda`), без номера раздела (например, `/dev/sda1`).

Комментарий

Прежде чем выполнять манипуляции с загрузчиком, создайте резервные копии своих данных. Если восстановить загрузку системы не удалось, то попробуйте переустановить GRUB с диска SystemRescue (см. рецепт 19.9).

Дополнительная информация

- Руководство GNU GRUB Manual (https://oreil.ly/zkwke).
- Глава 19.

ГЛАВА 3

Запуск, остановка, перезапуск и перевод Linux в спящий режим

В данной главе вы узнаете несколько способов остановки, запуска и перезапуска системы Linux, а также научитесь управлять спящими режимами. Вы познакомитесь как с устаревшими командами для выполнения этих действий, так и с новыми командами systemd.

Вы узнаете, как настроить автоматический запуск и выключение. Автоматическое выключение является хорошим способом напомнить себе, что пора заканчивать работу, и избавляет от необходимости помнить о выключении компьютера на ночь. Вы можете настроить автоматическое включение и выключение удаленного компьютера, чтобы иметь доступ к нему в рабочее время и не оставлять его включенным постоянно. Если ваши пользователи транжирят электроэнергию и не выключают свои компьютеры, то вы можете настроить их автоматическое выключение в нерабочее время.

«Комбинация из трех пальцев» Ctrl+Alt+Delete может пригодиться в тех ситуациях, когда какое-то приложение становится неуправляемым или когда нужно прервать запуск и перезагрузить компьютер. В графических средах рабочего стола для этой цели можно назначить более удобную комбинацию клавиш.

За последние десятилетия накопилось несколько устаревших команд выключения, во многом дублирующих друг друга: `shutdown`, `halt`, `poweroff` и `reboot`. Команда `shutdown` поддерживает дополнительные параметры для отключения

по времени с предупреждением всех пользователей, работающих в системе. Эти команды удобно использовать в сценариях, в сеансах SSH и вообще всегда, когда вы работаете в командной строке.



Привилегии root требуются не всегда

Раньше для выполнения команд выключения требовались привилегии суперпользователя root. Но времена меняются, и во многих современных дистрибутивах Linux эти команды больше не требуют привилегий root. Примеры в данной главе ориентированы на обычных непривилегированных пользователей. Если ваш конкретный дистрибутив Linux требует привилегий root, то он сообщит вам об этом.

В современных дистрибутивах Linux необходимость привилегий контролируется с помощью Polkit (ранее PolicyKit). См. man 8 polkit, чтобы узнать больше.

Но это еще не все, поскольку в дистрибутивах Linux с systemd (см. главу 4) старые классические команды не устанавливаются, а их имена даются символическим ссылкам на команду `systemctl`. В этом можно убедиться с помощью команды `stat`, как в следующем примере с `shutdown`:

```
$ stat /sbin/shutdown
  File: /sbin/shutdown -> /bin/systemctl
  Size: 14              Blocks: 0          IO Block: 4096    symbolic link
Device: 802h/2050d      Inode: 1177556      Links: 1
Access: (0777/lrwxrwxrwx)  Uid: ( 0/ root)    Gid: ( 0/ root)
```

В строке `File:` можно видеть, что `/sbin/shutdown` — это символическая ссылка на `/bin/systemctl`. Все имена устаревших команд — `/sbin/shutdown`, `/sbin/halt`, `/sbin/poweroff` и `/sbin/reboot` — являются символическими ссылками на `/bin/systemctl`. Символические ссылки с именами устаревших команд добавлены для обратной совместимости. В системах Linux без systemd вместо символических ссылок используются устаревшие выполняемые файлы.

В некоторых дистрибутивах Linux эти символические ссылки находятся в `/usr/sbin`, а не в `/sbin`. При использовании устаревших имен команд они ведут себя одинаково в системах с systemd и без systemd.

Кнопки управления питанием в графической среде рабочего стола можно настраивать и даже выбирать, какие кнопки будут отображаться и в каком порядке.

3.1. Выключение с помощью команды systemctl

Задача

Использовать команды `systemctl` для выключения и перезагрузки системы.

Решение

Остановить систему и выключить электропитание можно с помощью команды:

```
$ systemctl poweroff
```

Другой способ сделать то же самое:

```
$ systemctl shutdown
```

Перезагрузить:

```
$ systemctl reboot
```

Остановить систему, не выключая электропитание:

```
$ systemctl halt
```

Комментарий

Команды выключения `systemctl` не поддерживают ряд параметров, поддерживаемых старыми командами, что, впрочем, не имеет большого значения, поскольку старые команды предлагают много избыточных параметров. Однако есть одно существенное отличие: `systemctl shutdown` не имеет параметров для выключения по времени, которые поддерживаются старой командой `shutdown` (см. рецепт 3.2).

Дополнительная информация

- `man 8 systemd-halt.service`

3.2. Выключение, выключение по времени и перезагрузка с помощью команды shutdown

Задача

Выключить компьютер по времени, например через 10 минут или в конкретное время, и предупредить всех работающих в системе пользователей. Или выключить компьютер немедленно без всяких условий.

Решение

Команда `shutdown` всегда работает одинаково, будь то символическая ссылка на `systemctl` или устаревший выполняемый файл `shutdown`.

В следующих примерах показано, как выключить компьютер немедленно, через определенное количество минут или в определенное время, как отменить выключение, как остановить и перезагрузить систему.

Немедленное выключение без уведомления других пользователей:

```
$ shutdown -h now
```

Выключение через 10 минут с уведомлением пользователей:

```
$ shutdown -h +10
Shutdown scheduled for Sun 2021-05-23 11:04:43 PDT, use 'shutdown -c' to cancel.
```

Это сообщение могут увидеть другие пользователи системы, в зависимости от того, какую версию Linux они используют и открыт ли у них терминал:

```
Broadcast message from duchess@client4 on pts/4 (Sun 2021-05-24 10:54:43 PDT):

The system is going down for poweroff at Sun 2021-05-24 11:04:43 PDT!
```

Отмена выключения:

```
$ shutdown -c
```

Все пользователи в системе смогут увидеть следующее сообщение:

```
Broadcast message from duchess@client4 on pts/4 (Sun 2021-05-24 10:56:00 PDT):

The system shutdown has been cancelled
```

Послать свое сообщение можно так:

```
$ shutdown -h +6 "Time to stop working and go outside to play!"
```

Вместо количества минут, через которое следует выключить компьютер, можно указать точное время выключения в 24-часовом формате ЧЧ:ММ. Ниже представлен пример выключения компьютера в 22:15:

```
$ shutdown -h 22:15
```

Перезагрузка:

```
$ shutdown -r
```

Остановка без выключения электропитания:

```
$ shutdown -H
```

Запуск команды `shutdown` без параметров эквивалентен команде `shutdown -h +1`.

Комментарий

Команда `shutdown` отправляет сообщение другим пользователям только при использовании параметра `-h`, за исключением параметра `-h now`, и параметра `-k`. Такие сообщения называются *wall*-сообщениями, сокращенно от *write to all logged-in users* («написать всем вошедшим пользователям»). Разные дистрибутивы Linux по-разному поддерживают данную функцию, поэтому другие пользователи вашей конкретной системы Linux могут не видеть эти сообщения.

- `--help` выводит справочную информацию о поддерживаемых параметрах.
- `-H`, `--halt` останавливает систему без выключения электропитания компьютера; в этом случае вам придется нажать кнопку выключения на корпусе компьютера и удерживать ее некоторое время, чтобы выключить электропитание.
- `-P`, `--poweroff` останавливает систему и выключает электропитание компьютера.
- `-r`, `--reboot` останавливает систему и перезагружает компьютер.
- `-k` посылает wall-сообщение, не останавливая систему.
- `--no-wall` запрещает отправку wall-сообщения.

Дополнительная информация

- `man 8 shutdown`
- `man 1 wall`
- `man 8 systemd-halt.service`

3.3. Выключение и перезагрузка с помощью команд halt, reboot и poweroff

Задача

Вы научились пользоваться командой `shutdown` и теперь хотите знать, для чего нужны команды `halt`, `reboot` и `poweroff` и как ими пользоваться.

Решение

Все эти команды действуют почти одинаково.

Команда `halt` завершает работу системы, останавливая все службы и процессы и размонтируя файловые системы, но не выключает компьютер. Когда она будет выполнена, вы должны нажать и удерживать кнопку питания устройства, чтобы завершить выключение.

Команда `reboot` завершает работу системы и перезагружает компьютер.

Команда `poweroff` завершает работу системы и выключает компьютер:

```
$ halt
$ reboot
$ poweroff
```

Команды `halt` и `poweroff` могут также выполнить перезагрузку:

```
$ halt --reboot
$ poweroff --reboot
```

Комментарий

Если эти команды показались вам немного странными и избыточными, то вы совершенно правы. По мере старения программного обеспечения постепенно накапливается всякий хлам. Linux существует с 1991 года и начиналась как бесплатный клон системы Unix, родившейся в 1969 году. С тех пор прошло много лет, в течение которых разные разработчики писали код и добавляли свои любимые функции.

Команды `halt` и `poweroff` очень похожи и поддерживают одни и те же параметры:

- `--help` выводит справочную информацию о поддерживаемых параметрах;
- `--halt` останавливает систему без выключения электропитания компьютера (да, команды `halt` и `halt --halt` действуют совершенно одинаково);

- `-p`, `--poweroff` останавливает систему и выключает электропитание компьютера (да, команды `poweroff` и `poweroff --poweroff` действуют совершенно одинаково);
- `--reboot` останавливает систему и перезагружает компьютер;
- `-f`, `--force` выполняет принудительную остановку или выключение электропитания. В этом случае процедура остановки служб пропускается, все процессы прерываются и все файловые системы размонтируются или монтируются в режиме «только для чтения». Двойное использование параметра `--force` вызывает грубое прерывание системы; например, `poweroff -f -f` следует использовать, только когда компьютер не удается выключить обычным способом;
- `-w`, `--wtmp-only` не останавливает систему, а только делает соответствующую запись в `/var/log/wtmp`;
- `-d`, `--no-wtmp` предотвращает запись в `wtmp`.

Дополнительная информация

- `man 8 halt`
- `man 8 poweroff`
- `man 8 systemd-halt.service`

3.4. Перевод системы в спящий режим с помощью команды systemctl

Задача

Система Linux имеет systemd, и ее нужно перевести в спящий режим с помощью команды `systemctl`.

Решение

Команда `systemctl` поддерживает следующие режимы энергосбережения: *suspend*, *hibernate*, *hybrid-sleep* и *suspend-then-hibernate*.

Следующая команда переведет систему в режим приостановки (suspend):

```
$ systemctl suspend
```

В этом режиме текущий сеанс останется в ОЗУ, а аппаратное обеспечение будет переведено в состояние пониженного энергопотребления. Чтобы возобновить

работу системы, достаточно нажать любую клавишу, шевельнуть мышь или открыть крышку ноутбука.

Следующая команда переведет систему в спящий режим (hibernate):

```
$ systemctl hibernate
```

В этом режиме текущий сеанс сбрасывается на диск и электропитание компьютера выключается. Чтобы возобновить работу системы, нужно нажать кнопку включения электропитания и подождать одну-две минуты, пока сеанс восстановится.

Следующая команда переведет систему в гибридный спящий режим (hybrid-sleep):

```
$ systemctl hybrid-sleep
```

В этом режиме текущий сеанс остается в ОЗУ и сбрасывается на диск, затем все устройства выключаются, кроме ОЗУ. Если ваша система не поддерживает возможность электропитания ОЗУ отдельно от других устройств, то восстановление сеанса будет выполнено с диска. Чтобы возобновить работу системы, нужно нажать кнопку включения электропитания.

Следующая команда переведет систему в режим приостановки с последующим переводом в спящий режим (suspend-then-hibernate):

```
$ systemctl suspend-then-hibernate
```

В этом случае система сначала будет переведена в режим приостановки (suspend), а затем, спустя время, определяемое параметром `HibernateDelaySec=` в `/etc/systemd/sleep.conf`, — в спящий режим. Чтобы возобновить работу системы, нужно нажать кнопку включения электропитания.

Комментарий

См. подраздел «Комментарий» в рецепте 3.9, где подробно описываются разные режимы энергосбережения.

Графическая среда рабочего стола должна иметь кнопки для выбора того или иного режима энергосбережения и инструмент с графическим интерфейсом для управления такими событиями, как гашение и блокировка экрана, нажатие кнопки питания и действия с крышкой ноутбука и переход в спящий режим или выключение электропитания.

Управление электропитанием лучше всего реализовано на ноутбуках, но в некоторых дистрибутивах Linux может работать не так, как ожидается. На управление питанием влияют прошивка UEFI, особенности процессора, udev, расширен-

ный интерфейс конфигурации и питания (Advanced Configuration and Power Interface, ACPI), параметры компиляции ядра и, возможно, другие устройства и программы; очень многое зависит от того, как в вашем конкретном дистрибутиве Linux реализовано управление питанием. Загляните в документацию к вашему дистрибутиву.

Дополнительная информация

- `man 1 systemctl`
- `man 8 systemd-halt.service`

3.5. Надежная перезагрузка с помощью комбинации Ctrl+Alt+Delete

Задача

Требуется надежный, всегда действующий метод перезагрузки, даже в случае краха или выхода процессов из-под контроля.

Решение

Специально для этого в свое время была придумана «комбинация из трех пальцев» Ctrl+Alt+Delete. Последовательно нажмите и удерживайте эти три клавиши, и они, преодолев большинство проблем, перезагрузят вашу систему. В некоторых дистрибутивах Linux комбинация Ctrl+Alt+Delete отключена, но вы можете вновь включить ее.

Комбинация Ctrl+Alt+Delete управляется демоном systemd в консоли Linux. См. рецепт 3.6, чтобы узнать об управлении Ctrl+Alt+Delete в systemd.

Для систем без systemd см. подраздел «Комментарий» в этом рецепте.

Графические среды рабочего стола имеют свои инструменты для настройки поддержки Ctrl+Alt+Delete, независимо от systemd. Например, в диспетчере настроек Xfce4 есть модуль настройки клавиатуры (рис. 3.1); в GNOME — модуль Keyboard Settings (Настройки клавиатуры) в утилите GNOME Settings (Настройки GNOME).

Если вы предпочитаете другую комбинацию клавиш, то настройте ее. Вы можете даже назначить комбинацию из одной клавиши, хотя это увеличивает риск перезагрузки из-за случайного нажатия клавиши.

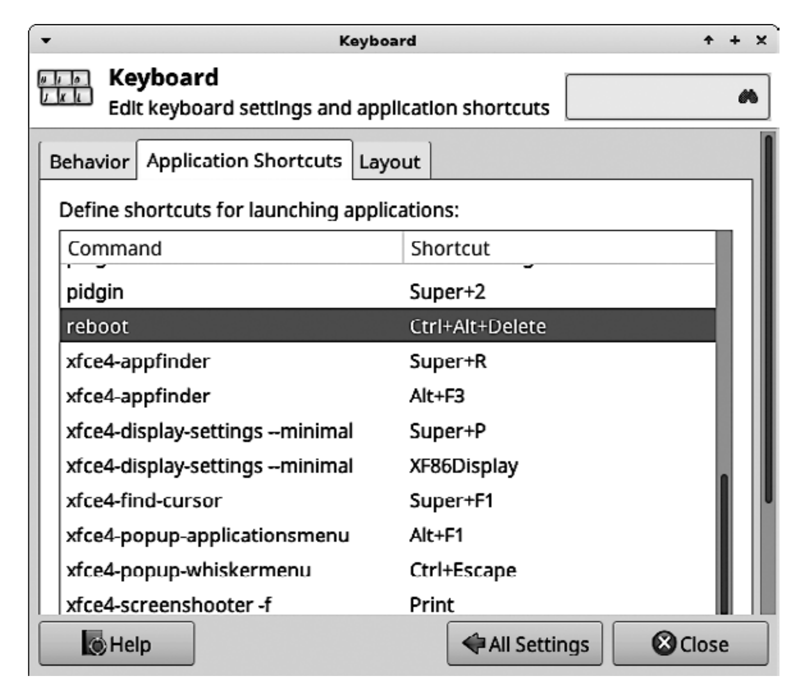


Рис. 3.1. Settings ▶ Keyboard ▶ Applications (Диспетчер настроек ▶ Клавиатура ▶ Комбинации клавиш), настройка горячих комбинаций клавиш в Xubuntu

Комментарий

В системах Linux без systemd комбинация Ctrl+Alt+Delete настраивается в файле `/etc/inittab`. Ниже представлен пример типичной конфигурации, взятый из MX Linux:

```
# Что делать по нажатии CTRL+ALT+DEL.
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
```

Последовательность цифр `12345` активирует комбинацию на уровнях запуска 1, 2, 3, 4 и 5. Параметр `-t1` означает «ждать одну секунду», `-a` вызывает `/etc/shutdown.allow`, а `-r` означает reboot — «перезагрузка». Настройте команду на выключение системы, добавив параметр `-h`, как при запуске `shutdown` из командной строки (см. рецепт 3.2). Чтобы отключить комбинацию Ctrl+Alt+Delete, закомментируйте строку с командой `shutdown`.

Обратите внимание, что параметры `-t1` и `-a` поддерживаются не всеми реализациями команды `shutdown`. Предыдущий пример взят из MX Linux. MX Linux поддерживает и систему инициализации Unix System V (SysV init), и systemd, и в меню загрузки предоставляется возможность выбрать между ними.

Поддержка Ctrl+Alt+Delete встроена в IBM PC BIOS/UEFI, и эта комбинация всегда должна перезагружать систему, если нажать ее до момента, когда GRUB запустит операционную систему.

Комбинация Ctrl+Alt+Delete была придумана инженером из IBM Дэвидом Брэдли (David Bradley) для IBM PC BIOS. Первоначально она предназначалась для использования исключительно разработчиками. По замыслу нажатие этой комбинации требовало двух рук, чтобы исключить любые случайности.

Затем Microsoft адаптировала его для вызова Диспетчера задач при первом нажатии и для перезагрузки при втором. Затем в Windows NT эта комбинация использовалась для доступа к экрану входа в Windows. Предположительно, это была мера безопасности, помогающая избежать обмана пользователей с помощью поддельных экранов входа, о существовании которых я и не подозревала, но это было давно. На YouTube сохранился видеоролик с забавным обменом мнениями между мистером Брэдли и Биллом Гейтсом об изобретении и использовании Ctrl+Alt+Delete, который, надеюсь, останется навсегда: Control-Alt-Delete: David Bradley & Bill Gates (https://oreil.ly/e83k6).

Дополнительная информация

- `man 7 systemd.special`

3.6. Включение, выключение и настройка комбинации Ctrl+Alt+Delete в консоли Linux

Задача

Поведением Ctrl+Alt+Delete в консоли Linux управляет systemd, и вам хотелось бы знать, как его настроить.

Решение

Вы можете проверить состояние комбинации Ctrl+Alt+Delete (включено или выключено) или изменить ее действие, чтобы вместо перезагрузки она выключала систему.

Файл модуля Ctrl+Alt+Delete — это не служба, а цель (target), поэтому он не запускается как демон. Если существует символическая ссылка `/etc/systemd/system/ctrl-alt-del.target`, то Ctrl+Alt+Delete активируется.

Следующий пример выключает и маскирует `ctrl-alt-del.target`:

```
$ sudo systemctl disable ctrl-alt-del.target
Removed /etc/systemd/system/ctrl-alt-del.target.

$ sudo systemctl mask ctrl-alt-del.target
Created symlink /etc/system/system/ctrl-alt-del.target → /dev/null.
```

Размаскирование и включение:

```
$ sudo systemctl unmask ctrl-alt-del.target
Removed /etc/systemd/system/ctrl-alt-del.target.

$ sudo systemctl enable ctrl-alt-del.target
Created symlink /etc/systemd/system/ctrl-alt-del.target →
/lib/systemd/system/reboot.target.
```

Изменения сразу же вступают в действие.

Изменить действие этой комбинации, чтобы вместо перезагрузки она выключала систему, можно, связав модуль `ctrl-alt-del.target` с модулем `poweroff.target`. Но прежде его нужно выключить, чтобы удалить существующую символическую ссылку, затем создать новую:

```
$ sudo systemctl disable ctrl-alt-del.target
Removed /etc/systemd/system/ctrl-alt-del.target.

$ sudo ln -s /lib/systemd/system/poweroff.target \
  /etc/systemd/system/ctrl-alt-del.target
```

Теперь она будет выключать систему вместо того, чтобы перезагружать ее.

Комментарий

Символические ссылки с помощью команды `stat` можно увидеть так:

```
$ stat /lib/systemd/system/ctrl-alt-del.target
  File: /lib/systemd/system/ctrl-alt-del.target -> reboot.target
  Size: 13              Blocks: 0          IO Block: 4096   symbolic link
Device: 802h/2050d      Inode: 136890      Links: 1
Access: (0777/lrwxrwxrwx)  Uid: ( 0/ root)   Gid: ( 0/ root)
```

Не изменяйте символические ссылки в `/lib/systemd/system/`. Вместо этого всегда создавайте новые символические ссылки в `/etc/systemd/system/`, чтобы ваше изменение не было затерто обновлениями системы.

Дополнительная информация

- `man 7 systemd.special`

3.7. Выключение по расписанию с помощью cron

Задача

Организовать выключение компьютера на ночь, чтобы можно было уйти и не беспокоиться об этом. Данный прием пригодится также, если ваши пользователи транжирят электроэнергию и никак не могут выработать привычку выключать свои компьютеры на ночь.

Решение

Используйте *cron* для выключения по расписанию. Например, добавьте следующую строку в `/etc/crontab` — это обеспечит выключение компьютера каждый вечер в 22:30 с предупреждением за 20 минут. Для редактирования `/etc/crontab` требуются привилегии root (для правки файла в этом примере использовался текстовый редактор nano):

```
$ sudo nano /etc/crontab
# мин час   дм  мес дн    пользователь команда
  10  22     *   *   *    root         /sbin/shutdown -h +20
```



Прежде чем делать это на рабочем компьютере, ознакомьтесь с правилами, установленными вашим работодателем. Если правила предусматривают запуск обновления и резервного копирования в ночное время, то, возможно, ваш компьютер должен оставаться включенным всю ночь.

Ниже представлен пример немедленного выключения системы в 23:00 в будние дни:

```
# мин час   дм  мес дн    пользователь  команда
  00  23     *   *  1-5   root          /sbin/shutdown -h now
```

Другой способ: использовать команду `crontab` с привилегиями root или через `sudo`:

```
$ sudo crontab -e
# мин час   дм  мес дн  команда
  00  23     *   *  1-5 /sbin/shutdown -h now
```

При редактировании с помощью команды `crontab` отсутствует поле `user`. Предыдущий пример открывает файл `crontab` пользователя root в режиме редактирования. Внесите необходимые изменения, сохраните — и все готово.

Не пытайтесь сохранить файл с другим именем. Во время редактирования с помощью команды `crontab` вы работаете с временным файлом, который автоматически получает имя `crontab` при сохранении. Он будет сохранен в `/var/spool/cron/crontabs`.

Комментарий

Файл `/etc/crontab` имеет поле «пользователь», поэтому в данном файле могут иметься записи, запускающие команды от имени любых пользователей, но только root может редактировать `/etc/crontab`. Пользователи, желающие определить свои расписания выполнения команд, должны использовать команду `crontab`, которая создает личные файлы `crontab` и не требует привилегий root.

Чтобы освоиться с полями в `/etc/crontab`, требуется практика (табл. 3.1), поэтому приведу еще несколько примеров и пояснений.

Таблица 3.1. Поля в файле crontab

Поле	Допустимые значения
мин (минуты)	0–59
час (часы)	0–23
дм (день месяца)	1–31
мес (месяц)	1–12
дн (день недели)	0–7

Звездочка (*) означает «все» или «любой».

Расписание выключения только в выходные дни выглядит следующим образом:

```
# выключать в 1:05 по субботам и воскресеньям
00 01   * * 7,0   root /sbin/shutdown -h +5
```

В cron есть одна хитрость: воскресенье может обозначаться как 0 или как 7. Причина уходит корнями в далекое прошлое, и я понятия не имею, почему такое положение вещей продолжает сохраняться. Вам нужно протестировать поведение своей версии cron, чтобы узнать, какая нумерация дней недели используется у вас; возможно, вам придется обозначать субботу и воскресенье как `6,7`:

```
00 01   * * 6,7   root /sbin/shutdown -h +5
```

Было бы неплохо иметь возможность использовать `sat,sun`, но названия дней недели можно применять только по одному и нельзя использовать в списках. Дни недели и месяцы обозначаются первыми тремя буквами: sat, sun, jan, feb. Регистр символов не имеет значения.

Вы можете использовать диапазоны: 1–4 означает 1, 2, 3 и 4.

Диапазоны и списки можно смешивать: 1, 3, 5, 6–10.

Можно указывать шаг в диапазонах:

- 10–23/2 означает: «каждый второй час между 10 и 23»[1];
- */2 в поле дня недели означает: «каждый второй день недели»;
- 2–6/2 равносильно 2, 4, 6.

Следующие строки можно использовать взамен первых пяти полей:

```
@reboot
@yearly
@annually
@monthly
@weekly
@daily
@midnight
@hourly
```

Дополнительная информация

- `man 8 cron`
- `man 1 crontab`
- `man 5 crontab`

3.8. Автоматическое включение по расписанию с помощью UEFI

Задача

Выключение по расписанию — замечательная возможность, но хотелось бы также иметь возможность включения по расписанию.

Решение

Считайте, что вам повезло, поскольку Linux поддерживает включение по расписанию. Есть три способа, которые можно попробовать: включение по локальной сети, включение по часам реального времени (RTC) или настройка UEFI компьютера, если в нем есть функция включения по расписанию.

[1] То есть команда будет выполнена в 10:00, 12:00, 14:00 и т. д.

Включение с помощью UEFI — самый надежный способ. На рис. 3.2 показан экран с расписанием включения на Lenovo ThinkPad.

Войдите в настройки BIOS/UEFI, нажав соответствующую клавишу Fn при запуске. В системах Dell, ASUS и Acer это обычно F2, а в Lenovo — F1. В других системах это может быть другая клавиша; например, в ряде систем используется клавиша Delete, так что загляните в документацию к своему компьютеру. Некоторые системы сообщают, какую клавишу нажать, на экране запуска. Часто трудно нажать клавишу в нужное время, поэтому начинайте и продолжайте нажимать ее сразу после нажатия кнопки питания, как если бы вы нажимали кнопку вызова лифта, чтобы он прибыл быстрее.

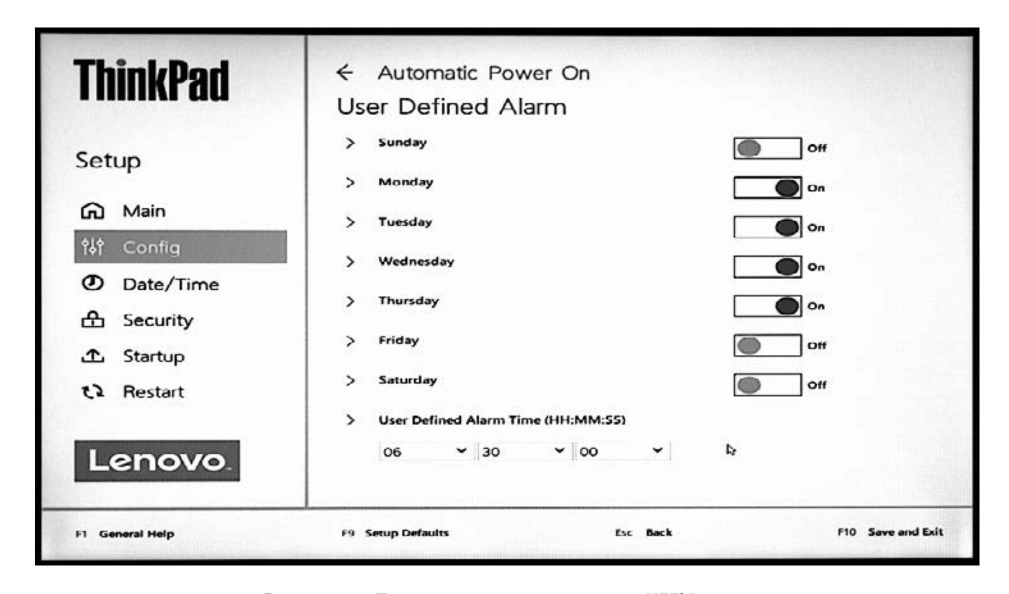


Рис. 3.2. Планирование включения в UEFI Lenovo

Если в вашей системе UEFI не поддерживает возможность включения по расписанию, то попробуйте вариант с включением по сети (см. рецепт 3.10, для которого требуется второе устройство, отправляющее сигнал включения) или вариант с включением по часам реального времени (см. рецепт 3.9).

Комментарий

Кратко обсудим назначение BIOS и UEFI. Сразу после включения питания первые инструкции, которые выполняет компьютер, извлекаются из прошивки базовой системы ввода-вывода (Basic Input Output System, BIOS) или единого

расширяемого микропрограммного интерфейса (Unified Extensible Firmware Interface, UEFI), хранящихся на материнской плате компьютера. BIOS — это старая система, которая используется с 1980 года. UEFI — ее современная замена, которая включает поддержку устаревшей BIOS, хотя когда-нибудь она будет удалена. Почти все компьютеры, выпущенные во второй половине 2000-х годов, имеют UEFI.

UEFI предлагает больше функций, чем старая BIOS, и напоминает небольшую операционную систему. Экраны настройки UEFI управляют порядком загрузки, загрузочными устройствами, параметрами безопасности, безопасной загрузкой, разгоном, отображением состояния оборудования и сети и предлагают множество других функций.

Дополнительная информация

- Рецепт 3.9.
- Рецепт 3.10.
- Рецепт 3.11.

3.9. Автоматическое включение по расписанию с помощью часов реального времени

Задача

Настроить включение по расписанию с помощью часов реального времени (real-time clock, RTC), поскольку ваша прошивка UEFI не поддерживает функцию включения по расписанию или просто потому, что вы так хотите.

Решение

Используйте команду `rtcwake`, которая должна присутствовать в вашей системе и устанавливается из пакета *util-linux*. Эта команда останавливает и запускает систему. Вы можете настроить вывод системы из спящего режима через указанный интервал, например через 1800 секунд, или в конкретное время и дату.

Часы реального времени (RTC) в вашей системе должны быть установлены на Всемирное координированное время (Coordinated Universal Time, UTC).

Когда `rtcwake` останавливает систему, он переводит ее в состояние сна ACPI. Загляните в `/sys/power/state`, чтобы узнать, какие состояния сна поддерживает ваша система. Система, откуда получен следующий пример, поддерживает только три из шести состояний сна ACPI:

```
$ cat /sys/power/state
freeze mem disk
```

В системах без systemd выполните команду `cat /proc/acpi/info`.

В системе, где был получен пример, показанный выше, можно попробовать три состояния сна, как показано ниже:

```
$ sudo rtcwake -m freeze -s 60
```

Параметр `-m` задает режим сна, а `-s` — количество секунд до повторного запуска системы. Запустив эту команду, вы увидите, как система переходит в спящий режим, затем возобновляет работу. При этом вы увидите сообщение об успехе или об ошибке.

Следующий пример выполняет тестовый запуск команды для перевода системы в состояние сна с пробуждением «завтра в 8:00»:

```
$ sudo rtcwake -n -m disk no -u -t $(date +%s -d "tomorrow 08:00")
rtcwake: wakeup from "disk" using /dev/rtc0 at Mon Nov 23 08:00:00 2021
```

Удалите параметр `-n`, чтобы выполнить команду не в тестовом режиме.

Ниже приводится хороший пример автоматизации выключения и включения системы в `/etc/crontab`. В 23:00 по будням команда `rtcwake` останавливает работу системы с сохранением состояния на диск и запускает ее спустя 8 часов:

```
# мин час   дм  мес дн    пользователь команда
  00  23     *   *  1-5   root         /usr/sbin/rtcwake -m disk -s 28800
```

Комментарий

Загляните в настройки BIOS/UEFI вашей системы и убедитесь, что аппаратные часы установлены по времени UTC. Если в настройках нет дополнительного параметра или кнопки для перевода часов реального времени в часовой пояс UTC, то измените время вручную, выставив текущее время UTC.

Пример, включающий параметры `-u -t +$(date +%s -d "tomorrow 08:00")`, преобразует время в формате от начала эпохи Unix в удобочитаемые значения. Время от начала эпохи Unix — это количество секунд, прошедших с полу-

ночи 1 января 1970 года по Всемирному координированному времени (UTC). Параметр `date +%s` возвращает текущее время от начала эпохи Unix, `-t` передает это время для преобразования, а `-u` указывает, что аппаратные часы выставлены по времени UTC.

Параметр `-n` сообщает команде `rtcwake`, что она не должна переводить систему в спящий режим, а только установить время пробуждения. Удалите параметр `-n`, чтобы перевести систему в состояние сна.

Включение по часам реального времени (RTC) — наименее надежный способ. В этом случае система переводится в состояние сна ACPI. ACPI — современный стандарт управления электропитанием и состояниями сна. Предполагается, что он не зависит от производителя и оборудования, но он очень сложен, и производители оборудования часто реализуют только часть его возможностей. Кроме того, разные дистрибутивы Linux реализуют его по-разному.

Существует шесть состояний сна ACPI, S0–S5:

- *S0* — система запущена, монитор может быть выключен, большинство устройств включено;
- *S1* — система приостановлена, процессоры простаивают, питание процессоров и ОЗУ включено;
- *S2* — питание процессоров выключено, кэши процессоров сброшены в ОЗУ;
- *S3* также называется режимом ожидания, сна, приостановки с сохранением состояния в ОЗУ. Данные могут не сбрасываться на диск;
- *S4* — режим ожидания, приостановки с сохранением состояния на диск. Все содержимое ОЗУ записывается на диск, и питание системы выключается;
- *S5* подобно полному выключению системы, за исключением того, что кнопка питания и периферийные устройства, такие как клавиатура, сетевой интерфейс и устройства USB, остаются под напряжением.

Ядро Linux способно поддерживать до четырех состояний, а дистрибутивы — разное количество из этих четырех.

Дополнительная информация

- `man 8 cron`
- `man 8 rtcwake`
- Сайт Time and Date (https://timeanddate.com).

3.10. Настройка удаленного включения по сети с помощью проводного Ethernet

Задача

Настроить включение компьютера с помощью удаленного вызова по сети, поскольку прошивка UEFI вашего компьютера не поддерживает включение по расписанию, или она слишком проста для ваших нужд, или нужна возможность отправки сигнала включения в произвольное время. Этот рецепт описывает настройку включения компьютера, находящегося в той же сети, что и устройство, которое предполагается использовать для отправки сигнала включения, при этом целевой компьютер должен быть подключен к сети через проводной интерфейс Ethernet.

Решение

Настройте компьютер на прием запросов на включение, а затем используйте второе устройство, например другой компьютер, смартфон или Raspberry Pi, для отправки сигнала включения, который называется *волшебным пакетом (magic packet)*. На самом деле ничего волшебного здесь нет. Это всего лишь специализированный пакет для включения удаленного компьютера, находящегося в спящем режиме. (Да, мне тоже жаль, что это не настоящее волшебство.)

Для начала зайдите в настройки UEFI вашей системы и найдите параметр, разрешающий включение по сети (Wake-on-LAN).



Важно отключить все параметры, разрешающие загрузку по протоколу удаленной загрузки PXE (Preboot eXecution Environment). Если загрузка PXE включена и в вашей сети есть PXE-сервер (который загружается с сервера сетевой установки), то может получиться так, что ваш компьютер будет включен по команде PXE и установит новый образ, стерев существующую установку.

Затем выйдите и загрузите систему. Установите пакеты *wakeonlan* и *ethtool*.

Узнайте имя вашего интерфейса Ethernet (в этом примере — `enp0s25`) и с помощью ethtool проверьте, поддерживает ли он включение по сети. Для ясности следующий вывод сокращен:

```
$ ip addr show
2: enp0s25: <BROADCAST,MULTICAST,UP,LOWER_UP> state UP 0
```

```
    link/ether 9c:ef:d5:fe:8f:20 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.97/24 brd 192.168.1.255 scope global dynamic
    [...]

$ sudo ethtool enp0s25 | grep -i wake-on
        Supports Wake-on: pumbg
        Wake-on: g
```

Запишите MAC-адрес интерфейса. В предыдущем примере выполнения команды `ip` MAC-адрес находится в строке `ether` и имеет значение `9c:ef:d5:fe:8f:20`. У вас будет другой адрес, поскольку MAC-адреса уникальны.

`Supports Wake-on: pumbg` — волшебная фраза, подтверждающая, что ваш интерфейс имеет необходимую поддержку, обозначенную переключателем `g`. Вторая строка `Wake-on: g` сообщает, что эта поддержка уже включена. Если это не так, то включите ее:

```
$ sudo ethtool -s enp0s25 wol g
```

Если после перезагрузки системы поддержка выключится, то добавьте следующую запись в `/etc/crontab` для запуска этой команды после каждой загрузки:

```
$ @reboot root /usr/bin/ethtool -s enp0s25 wol g
```

Выключите компьютер и со второго устройства в той же сети отправьте команду для его включения, указав MAC-адрес Ethernet-интерфейса целевого компьютера:

```
$ /usr/bin/wakeonlan 9c:ef:d5:fe:8f:20
```

Если целевой компьютер и второе устройство находятся в одной сети, но в разных подсетях, то укажите широковещательный адрес для подсети с вашим целевым компьютером:

```
$ /usr/bin/wakeonlan -i 192.168.44.255 9c:ef:d5:fe:8f:20
```

Комментарий

Включение по сети (Wake-on-LAN) — это стандарт Ethernet удаленного включения компьютера путем отправки ему сигнала пробуждения по сети. Именем `wakeonlan` называются команда и пакет в большинстве дистрибутивов Linux.

При выключении в действительности компьютер не выключается, а переходит в режим низкого энергопотребления и может принимать и реагировать на сигнал включения.

Команда `wakeonlan` посылает волшебный пакет в порт UDP 9. Когда волшебный пакет отправляется на широковещательный адрес сети, то его получат все хосты в сети. MAC-адрес гарантирует, что включится только хост с этим адресом.

Целевой компьютер включается так же, как если бы вы нажали кнопку питания.

Дополнительная информация

- `man 1 wakeonlan`
- `man 8 ethtool`
- Рецепт 3.8.
- Рецепт 3.9.
- Рецепт 3.11.

3.11. Настройка удаленного включения через Wi-Fi (WoWLAN)

Задача

Включить удаленный компьютер через беспроводной интерфейс (Wake-on-Wireless LAN, или WoWLAN).

Решение

Этот рецепт предназначен для включения компьютера, находящегося в той же сети, что и устройство, которое используется для отправки сигнала включения.

Компьютер должен иметь беспроводной интерфейс, встроенный в материнскую плату или подключенный к шине PCI. Этот рецепт не будет работать с интерфейсом USB, поскольку на шину USB не подается питание, когда устройство выключено.

Сначала войдите в настройки UEFI машины, которую нужно включать удаленно, и установите все настройки Wake-on-LAN.



Важной мерой предосторожности является отключение всех параметров, разрешающих загрузку по протоколу удаленной загрузки PXE (Preboot eXecution Environment). Если загрузка PXE включена и в вашей сети есть PXE-сервер (который загружается с сервера сетевой установки), то может получиться так, что ваш компьютер будет включен по команде PXE и установит новый образ, затерев существующую установку.

Затем выйдите и загрузите систему. Установите команду `iw` и используйте ее для вывода списка всех имеющихся беспроводных устройств:

```
$ iw dev
phy#0
        Interface wlxcc3fd5fe014c
                ifindex 3
                wdev 0x1
                addr 9c:bf:25:fe:0e:7c
                ssid accesspointe
                type managed
                channel 11 (2462 MHz), width: 20 MHz, center1: 2462 MHz
                txpower 20.00 dBm
```

Если таких устройств несколько, запросите то, которое будете использовать. В следующем примере показан беспроводной интерфейс, не поддерживающий WoWLAN:

```
$ iw phy0 wowlan show
command failed: Operation not supported (-95)
```

А в этом примере показан интерфейс, поддерживающий WoWLAN, но данная функция в нем отключена:

```
$ iw phy0 wowlan show
WoWLAN is disabled
```

Включите поддержку WoWLAN:

```
$ sudo iw phy0 wowlan enable magic-packet
WoWLAN is enabled:
  * wake up on magic packet
```

Команда `iw dev` выводит MAC-адрес, который нужно использовать на другом устройстве, чтобы сформировать волшебный пакет:

```
$ /usr/bin/wakeonlan 9c:bf:25:fe:0e:7c
```

Чтобы включить компьютер в той же сети, но в другой подсети, можно отправить волшебный пакет на широковещательный адрес подсети:

```
$ /usr/bin/wakeonlan -i 192.168.44.255 9c:bf:25:fe:0e:7c
```

Комментарий

Этот способ особенно хорош, когда удаленным компьютером является ноутбук, поскольку почти все современные ноутбуки имеют встроенные беспроводные сетевые интерфейсы, которые часто поддерживают широкий круг возможностей.

Настольные компьютеры обычно не оснащаются встроенными беспроводными интерфейсами, поэтому, выбирая беспроводной адаптер PCI/PCIe при покупке, обращайте внимание на поддержку WoWLAN и Linux.

Дополнительная информация

- `man 8 iw`
- `man 1 wakeonlan`
- Рецепт 3.8.
- Рецепт 3.9.
- Рецепт 3.10.

ГЛАВА 4

Управление службами с помощью systemd

Каждый раз, когда вы включаете свой компьютер с Linux, система инициализации запускает множество процессов, от нескольких десятков до сотен, в зависимости от настроек системы. Увидеть это можно на экране запуска (рис. 4.1; нажмите клавишу Esc, чтобы выключить графический экран запуска и увидеть сообщения о запуске служб).

```
[  OK  ] Reached target Local Encrypted Volumes.
[  OK  ] Listening on Syslog Socket.
         Starting Journal Service...
[  OK  ] Mounted Kernel Debug File System.
[  OK  ] Mounted POSIX Message Queue File System.
[  OK  ] Mounted Huge Pages File System.
[  OK  ] Started Load Kernel Modules.
[  OK  ] Started Create list of required static device nodes for the current kernel.
[  OK  ] Started Remount Root and Kernel File Systems.
         Starting udev Coldplug all Devices...
         Starting Create Static Device Nodes in /dev...
         Starting Apply Kernel Variables...
[  OK  ] Started Journal Service.
[  OK  ] Started Create Static Device Nodes in /dev.
[  OK  ] Started Apply Kernel Variables.
[  OK  ] Stopped Entropy Daemon based on the HAVEGE algorithm.
[  OK  ] Started Entropy Daemon based on the HAVEGE algorithm.
         Starting udev Kernel Device Manager...
[  OK  ] Started udev Coldplug all Devices.
[  OK  ] Started Setup Virtual Console.
[  OK  ] Started udev Kernel Device Manager.
[    3.630490] pcieport 0000:00:02.6: pciehp: Failed to check link status
[    3.662015] input: Power Button as /devices/LNXSYSTM:00/LNXPWRBN:00/input/input4
[    3.667655] ACPI: Power Button [PWRF]
[  OK  ] Created slice system-qemu\x2dga.slice.
         Starting Setup Virtual Console...
```

Рис. 4.1. Сообщения во время загрузки Linux

В прежние времена для запуска процессов во время загрузки использовалась система инициализации Unix System V (SysV init), BSD init и Linux Standard Base (LSB) init. Наиболее распространенной была SysV init. Те времена остались в прошлом, и сейчас используется новая замечательная система инициализации systemd. Она была взята на вооружение всеми основными дистрибутивами Linux, хотя, конечно, есть несколько дистрибутивов, которые продолжают прибегать к устаревшим системам инициализации.

В этой главе вы проверите, использует ли ваш дистрибутив Linux систему инициализации systemd, узнаете, что такое процессы, потоки, службы и демоны и как с помощью systemd управлять службами: запускать, останавливать, включать, отключать и проверять статус. Вдобавок вы познакомитесь с командой `systemctl`, диспетчером служб и системы systemd.

Система systemd проектировалась для поддержки современных сложных серверных и настольных систем и обладает гораздо более широкими возможностями, чем прежние системы инициализации. Она обеспечивает полноценное управление службами от запуска до завершения системы, запуск процессов при загрузке и после загрузки и завершение служб, когда они не нужны. Система systemd управляет такими функциями, как поддержка системного журнала, автоматическое монтирование файловых систем, автоматическое разрешение зависимостей служб, управление устройствами, управление сетевыми подключениями, управление входом в систему и множество других задач.

Список возможностей выглядит внушительно, но только пока вы не узнаете, что всю работу на компьютере выполняют процессы и все эти возможности и раньше поддерживались большим набором других программ. Система systemd просто объединила все это в интегрированный программный пакет, который должен работать одинаково во всех системах Linux, хотя, как всегда, разные дистрибутивы Linux могут иметь незначительные различия, такие как местоположение файлов и названия служб. Имейте в виду, что ваш конкретный дистрибутив Linux может отличаться от примеров в этой главе.

Система systemd пытается уменьшить время загрузки и более эффективно распределять системные ресурсы, запуская процессы одновременно и параллельно и инициализируя только необходимые службы, откладывая запуск других служб на период после загрузки. Служба, зависящая от других служб, больше не должна ждать, пока они запустятся, поскольку ей достаточно получить доступ к сокету Unix. Рецепт 4.9 показывает, как найти процессы, замедляющие запуск вашей системы.

Двоичные файлы systemd написаны на C, что обеспечивает повышенную производительность. В прежних системах инициализации использовалось множество сценариев на языке командной оболочки, а, как известно, любой компилирующий язык работает быстрее, чем сценарии оболочки.

Система systemd обратно совместима с SysV init. Большинство дистрибутивов Linux сохраняют устаревшие файлы конфигурации и сценарии SysV, включая `/etc/inittab` и каталоги `/etc/rc.d/` и `/etc/init.d/`. Если служба не имеет файла конфигурации в формате systemd, то systemd попытается отыскать конфигурационный файл в формате SysV. Кроме того, systemd обратно совместима с системой инициализации Linux Standard Base (LSB) init.

Служебные файлы systemd меньше и понятнее, чем файлы SysV init. Сравните файл из системы SysV init для запуска sshd с его аналогом из systemd. Ниже представлен фрагмент файла инициализации `/etc/init.d/ssh` из MX Linux:

```
#! /bin/sh

### BEGIN INIT INFO
# Provides:             sshd
# Required-Start:       $remote_fs $syslog
# Required-Stop:        $remote_fs $syslog
# Default-Start:        2 3 4 5
# Default-Stop:
# Short-Description:    OpenBSD Secure Shell server
### END INIT INFO

set -e

# /etc/init.d/ssh: start and stop the OpenBSD "secure shell(tm)" daemon

test -x /usr/sbin/sshd || exit 0

umask 022

if test -f /etc/default/ssh; then
[...]
```

Всего этот файл содержит 162 строки. А вот полный файл `/lib/systemd/system/ssh.service` из systemd в Ubuntu 20.04:

```
[Unit]
Description=OpenBSD Secure Shell server
Documentation=man:sshd(8) man:sshd_config(5)
After=network.target auditd.service
ConditionPathExists=!/etc/ssh/sshd_not_to_be_run

[Service]
EnvironmentFile=-/etc/default/ssh
ExecStartPre=/usr/sbin/sshd -t
ExecStart=/usr/sbin/sshd -D $SSHD_OPTS
ExecReload=/usr/sbin/sshd -t
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
```

```
RestartPreventExitStatus=255
Type=notify
RuntimeDirectory=sshd
RuntimeDirectoryMode=0755

[Install]
WantedBy=multi-user.target
Alias=sshd.service
```

Даже ничего не зная о systemd и не читая документацию, вы легко поймете, что должен делать этот файл.

Подробное введение в systemd вы найдете в статье *Rethinking PID 1* (https://oreil.ly/dFz4K)[1], написанной Леннартом Поттерингом (Lennart Poettering) — одним из ее изобретателей и разработчиков. В статье подробно описываются причины создания новой системы инициализации, ее архитектура, преимущества и то, как она использует новые возможности ядра Linux вместо дублирования существующих функций.

4.1. Проверка использования systemd в вашем дистрибутиве Linux

Задача

Узнать, использует ли ваш дистрибутив Linux систему инициализации systemd.

Решение

Поищите каталог `/run/systemd/system/`. Если он существует, значит, ваш дистрибутив использует систему инициализации systemd.

Комментарий

Каталог `/run/systemd/` может также присутствовать в системе, когда дистрибутив поддерживает несколько систем инициализации, но если отсутствует каталог `/run/systemd/system/`, то это говорит о том, что systemd не является активной системой инициализации.

Есть несколько других способов узнать, какая система инициализации используется вашим дистрибутивом. Попробуйте запросить статус файла `/sbin/init`.

[1] Перевод статьи на русский язык можно найти по адресу http://tux-the-penguin.blogspot.com/2010/09/systemd.html.

Первоначально это был выполняемый файл системы SysV, но в настоящее время большинство дистрибутивов Linux сохраняют данное имя и присваивают его символической ссылке, указывающей на выполняемый файл systemd. Следующий пример подтверждает, что в данном дистрибутиве Linux используется система инициализации systemd:

```
$ stat /sbin/init
File: /sbin/init -> /lib/systemd/systemd
[...]
```

В системах с SysV init этот файл не является символической ссылкой:

```
$ stat /sbin/init
File: /sbin/init
[...]
```

Псевдофайловая система `/proc` является интерфейсом к ядру Linux и содержит информацию о текущем состоянии выполняющейся системы. Она называется псевдофайловой системой, поскольку существует только в оперативной памяти. В данном примере `/proc/1/exe` — это символическая ссылка на выполняемый файл systemd:

```
$ sudo stat /proc/1/exe
File: /proc/1/exe -> /lib/systemd/systemd
[...]
```

В системе с SysV эта ссылка указывает на файл `init`:

```
$ sudo stat /proc/1/exe
File: /proc/1/exe -> /sbin/init
[...]
```

Файл `/proc/1/comm` сообщает название активной системы инициализации:

```
$ cat /proc/1/comm
systemd
```

В системе с SysV этот файл содержит название `init`:

```
$ cat /proc/1/comm
init
```

Команда для запуска процесса с идентификатором 1 (Process Identifier, PID) — это выполняемый файл системы инициализации. Процесс с PID 1 — первый процесс, запущенный во время загрузки, который затем запускает все остальные процессы. Увидеть его можно также с помощью команды `ps`:

```
$ ps -p 1
  PID TTY          TIME CMD
    1 ?        00:00:00 systemd
```

В системе с SysV вывод этой команды выглядит так:

```
$ ps -p 1
  PID TTY          TIME CMD
    1 ?        00:00:00 init
```

Более подробная информация о PID 1 приводится в рецепте 4.2.

Разные дистрибутивы Linux по-разному поддерживают systemd. Ее используют большинство основных дистрибутивов Linux, включая Fedora, Red Hat, CentOS, openSUSE, SUSE Linux Enterprise, Debian, Ubuntu, Linux Mint, Arch, Manjaro, Elementary и Mageia.

Некоторые популярные дистрибутивы не поддерживают systemd или включают ее, но не используют по умолчанию, в том числе: Slackware, PCLinuxOS, Gentoo Linux, MX Linux и antiX.

Дополнительная информация

- Сайт Distrowatch (https://distrowatch.com), содержащий информацию о сотнях дистрибутивов Linux.
- `man 5 proc`
- `man 1 pstree`
- `man 1 ps`

4.2. Процесс с PID 1 — родоначальник всех процессов

Задача

Поближе познакомиться со службами и процессами в Linux.

Решение

Процесс с идентификатором PID 1 — родоначальник всех процессов в системах Linux. Это первый процесс, который запускается в системе и запускает все остальные процессы.

Процессы — это экземпляр запущенной программы. Каждая задача в системе Linux представлена процессом. Процессы могут создавать независимые копии самих себя, то есть могут *ветвиться*. Ответвленные копии называются *потомками*, а оригинал — *родителем* или *предком*. Каждый потомок получает свой

уникальный идентификатор PID и набор системных ресурсов, таких как процессорное время и память. *Потоки выполнения* — это легковесные процессы, выполняющиеся параллельно и использующие системные ресурсы совместно со своими предками.

Некоторые процессы работают в фоновом режиме и не взаимодействуют с пользователями. В Linux такие процессы называются *службами* или *демонами*, и их имена обычно заканчиваются буквой D, например: httpd, sshd и systemd.

Всякая система Linux сначала запускает процесс с PID 1, который затем запускает все остальные процессы. Получить список всех запущенных процессов можно с помощью команды `ps`:

```
$ ps -ef
UID        PID  PPID  C STIME TTY          TIME CMD
root         1     0  0 10:06 ?        00:00:01 /sbin/init splash
root         2     0  0 10:06 ?        00:00:00 [kthreadd]
root         3     2  0 10:06 ?        00:00:00 [rcu_gp]
root         4     2  0 10:06 ?        00:00:00 [rcu_par_gp]
[...]
```

Команда `pstree` организует этот массив информации в виде древовидной диаграммы. В следующем примере показаны все процессы, их потомки, идентификаторы PID и потоки, заключенные в фигурные скобки:

```
$ pstree -p
systemd(1)─┬─ModemManager(925)─┬─{ModemManager}(944)
           │                   └─{ModemManager}(949)
           ├─NetworkManager(950)─┬─dhclient(1981)
           │                     ├─{NetworkManager}(989)
           │                     └─{NetworkManager}(991)
           ├─accounts-daemon(927)─┬─{accounts-daemon}(938)
           │                      └─{accounts-daemon}(948)
           ├─acpid(934)
           ├─agetty(1103)
           ├─avahi-daemon(953)───avahi-daemon(970)
[...]
```

Полный вывод `pstree` довольно велик. С помощью этой команды можно также просмотреть дерево предков, потомков и потоков отдельного процесса, указав его PID, как показано в следующем примере для текстового редактора Kate:

```
$ pstree -sp 5193
systemd(1)───kate(5193)─┬─bash(5218)
                        ├─{kate}(5195)
                        ├─{kate}(5196)
                        ├─{kate}(5197)
                        ├─{kate}(5198)
                        ├─{kate}(5199)
[...]
```

Здесь видно, что `systemd(1)` является предком Kate, `bash(5218)` — потомком, а все процессы в фигурных скобках — это потоки.

Комментарий

Процессы всегда находятся в одном из нескольких состояний, которые меняются в зависимости от активности системы. В следующем примере вывода команды `pstree` отображаются поля PID, `user` (пользователь), `state` (состояние) и `command` (команда):

```
$ ps -eo pid,user,stat,comm
  PID USER       STAT COMMAND
    1 root       Ss   systemd
    2 root       S    kthreadd
   32 root       I<   kworker/3:0H-kb
   68 root       SN   khugepaged
11222 duchess    Rl   konsole
```

- `R` — процесс выполняется или ожидает своей очереди на выполнение.
- `l` — многопоточный процесс.
- `S` — процесс находится в состоянии прерываемого ожидания; процесс ждет наступления некоторого события.
- `s` — лидер сеанса. Сеансы — это группы родственных процессов, действующих как единое целое.
- `I` — поток бездействия ядра.
- `<` — высокий приоритет.
- `N` — низкий приоритет.

Существует еще несколько редко используемых состояний; прочитать о них вы сможете в руководстве `man 1 ps`.

Дополнительная информация

- Рецепт 4.7.
- `man 5 proc`
- `man 1 pstree`
- `man 1 ps`

4.3. Вывод списка служб и их состояний с помощью команды systemctl

Задача

Перечислить все службы, установленные в системе, и вывести их состояние: запущены они или находятся в состоянии ошибки.

Решение

Использовать `systemctl` — команду-диспетчер systemd. Запустите ее без параметров, чтобы получить список всех загруженных модулей. Модуль systemd — это любой пакет процессов, связанных друг с другом, определенных в файле конфигурации модуля и управляемых системой systemd:

```
$ systemctl
```

Эта команда выведет гигантский объем информации: 177 активных загруженных модулей в моей тестовой системе с полными названиями модулей, информацией о статусе и подробными описаниями. Переадресуйте вывод в текстовый файл, чтобы потом было проще исследовать его:

```
$ systemctl > /tmp/systemctl-units.txt
```

Побалуйте себя дополнительной информацией, перечислив все модули, активные и неактивные:

```
$ systemctl --all
```

В моей тестовой системе эта команда вывела список из 349 модулей, включая *отсутствующие* и *неактивные*. Сколько всего файлов модулей? В следующем примере показаны пять файлов из 322:

```
$ systemctl list-unit-files
UNIT FILE                                  STATE
proc-sys-fs-binfmt_misc.automount          static
-.mount                                    generated
mount                                      generated
dev-hugepages.mount                        static
home.mount                                 generated
[...]
322 unit files listed.
```

Нас интересуют файлы служб, поскольку пользователи и администраторы Linux в основном взаимодействуют с ними и редко с другими типами файлов. Сколько всего таких файлов установлено? Посмотрим:

```
$ systemctl list-unit-files --type=service
UNIT FILE                                   STATE
accounts-daemon.service                     enabled
acpid.service                               disabled
alsa-state.service                          static
alsa-utils.service                          masked
anacron.service                             enabled
[...]
212 unit files listed.
```

В предыдущем примере можно видеть четыре наиболее часто встречающихся статуса служб: `enabled` (включена), `disabled` (выключена), `static` (статическая) и `masked` (замаскированная).

Список всех включенных служб:

```
$ systemctl list-unit-files --type=service --state=enabled
UNIT FILE                                   STATE
accounts-daemon.service                     enabled
anacron.service                             enabled
apparmor.service                            enabled
autovt@.service                             enabled
avahi-daemon.service                        enabled
[...]
62 unit files listed.
```

Список всех выключенных служб:

```
$ systemctl list-unit-files --type=service --state=disabled
UNIT FILE                                   STATE
acpid.service                               disabled
brltty.service                              disabled
console-getty.service                       disabled
mariadb@.service                            disabled
[...]
12 unit files listed.
```

Список всех статических служб:

```
$ systemctl list-unit-files --type=service --state=static
UNIT FILE                                   STATE
alsa-restore.service                        static
alsa-state.service                          static
apt-daily-upgrade.service                   static
apt-daily.service                           static
[...]
106 unit files listed.
```

Список всех замаскированных служб:

```
$ systemctl list-unit-files --type=service --state=masked
UNIT FILE                                  STATE
alsa-utils.service                         masked
bootlogd.service                           masked
bootlogs.service                           masked
checkfs.service                            masked
[...]
36 unit files listed.
```

Комментарий

Файлы служб модулей находятся в `/usr/lib/systemd/system/` или `/lib/systemd/system/`, в зависимости от того, куда их помещает дистрибутив Linux. Это текстовые файлы, которые можно читать.

- Статус `enabled` показывает, что служба доступна и управляется системой systemd. Когда служба включена, systemd создает символическую ссылку в `/etc/systemd/system/` на файл модуля в `/lib/systemd/system/`. Такую службу можно запустить, остановить, перезапустить и отключить с помощью команды `systemctl`.



Включение службы не приводит к ее запуску, а отключение — к остановке (см. рецепт 4.6).

- Статус `disabled` означает, что в `/etc/systemd/system/` нет символической ссылки, и эта служба не запускается автоматически при загрузке (но ее можно запустить и остановить вручную).
- Статус `masked` говорит о том, что служба ссылается на `/dev/null/`. Она полностью отключена, и ее нельзя запустить никаким способом.
- Статус `static` означает, что файл модуля является зависимостью для других файлов модуля и не может быть запущен или остановлен пользователем.

Ниже представлены и другие более редкие статусы служб, которые можно увидеть:

- `indirect` присваивается службам, которые не предназначены для управления пользователями, но могут управляться другими службами;
- `generated` сообщает, что служба была преобразована из конфигурационного файла системы инициализации, отличной от systemd, — SysV init или LSB init.

Дополнительная информация

- `man 1 systemctl`

4.4. Определение состояния выбранных служб

Задача

Узнать состояние одной или нескольких конкретных служб.

Решение

Команда `systemctl status` выводит небольшое подмножество полезной информации о состоянии службы. В следующем примере определяется состояние службы CUPS. Служба CUPS (Common Unix Printing System — общая система печати Unix) должна присутствовать во всех системах Linux:

```
$ systemctl status cups.service
● cups.service - CUPS Scheduler
     Loaded: loaded (/lib/systemd/system/cups.service; enabled; vendor preset:
             enabled)
     Active: active (running) since Sun 2021-11-22 11:01:48 PST; 4h 17min ago
TriggeredBy: ● cups.path
             ● cups.socket
       Docs: man:cupsd(8)
   Main PID: 1403 (cupsd)
      Tasks: 2 (limit: 18760)
     Memory: 3.8M
     CGroup: /system.slice/cups.service
             ├─1403 /usr/sbin/cupsd -l
             └─1421 /usr/lib/cups/notifier/dbus dbus://

Nov 22 11:01:48 host1 systemd[1]: Started CUPS Scheduler.
```

Чтобы запросить состояние нескольких служб, достаточно перечислить их через пробел:

```
$ systemctl status mariadb.service bluetooth.service lm-sensors.service
```

Комментарий

В этом небольшом фрагменте вывода содержится много полезной информации (рис. 4.2).

Круглый маркер рядом с именем службы — это индикатор состояния. Он отображается разными цветами в большинстве терминалов. Белый — *неактивное* состояние. Красный — состояние *сбоя* или *ошибки*. Зеленый — служба *активна*,

запускается или *перезапускается*. Остальная информация в выводе интерпретируется следующим образом.

- `Loaded` сообщает, загружен ли файл модуля в память и полный путь к нему, текущий статус службы (см. список статусов в разделе «Комментарий» в рецепте 4.3) и статус, предопределенный производителем (`vendor preset:`), который сообщает, должна ли, по мнению производителя, запускаться эта служба во время загрузки. Когда служба выключена, это означает, что по умолчанию производитель не предполагал запускать ее при загрузке. Этот параметр лишь отражает предпочтения производителя и не сообщает, включена ли служба в настоящее время.
- `Active` сообщает, активна ли служба и как долго она находится в этом состоянии.
- `Process` сообщает идентификаторы PID процессов, команды и имена демонов.
- `Main PID` — это номер процесса для сегмента контрольной группы.
- `Tasks` сообщает количество задач, запущенных службой. Задачи — это идентификаторы процессов PID.
- `CGroup` показывает, к какому сегменту модуля принадлежит служба и его PID. Существует три сегмента модулей по умолчанию: `user.slice`, `system.slice` и `machine.slice`.

  Контрольные группы Linux (cgroups) — это наборы связанных процессов и всех их потомков. *Сегмент* в systemd — это часть контрольной группы, и каждый сегмент управляет определенной группой процессов. Запустите `systemctl status`, чтобы увидеть иерархию контрольных групп.

  По умолчанию службы и области видимости модулей сгруппированы в `/lib/systemd/system/system.slice`.

  Пользовательские сеансы сгруппированы в `/lib/systemd/system/user.slice`.

  Виртуальные машины и контейнеры, зарегистрированные в systemd, сгруппированы в `/lib/systemd/system/machine.slice`.

```
duchess@client4:~$ systemctl status cups.service
● cups.service - CUPS Scheduler
     Loaded: loaded (/lib/systemd/system/cups.service; enabled; vendor p>
     Active: active (running) since Sun 2021-04-18 10:21:28 PDT; 1h 30mi>
TriggeredBy: ● cups.path
             ● cups.socket
       Docs: man:cupsd(8)
   Main PID: 991 (cupsd)
      Tasks: 1 (limit: 18755)
     Memory: 2.6M
     CGroup: /system.slice/cups.service
             └─991 /usr/sbin/cupsd -l
```

Рис. 4.2. Команда systemctl status вывела информацию о службе CUPS

Остальные строки — это самые последние записи в журнале, полученные командой `journalctl` — диспетчером журналов в systemd.

Дополнительная информация

- `man 1 systemctl`
- `man 5 systemd.slice`
- `man 1 journalctl`
- Документация с описанием контрольных групп ядра (https://oreil.ly/FfUb3).

4.5. Запуск и остановка служб

Задача

Научиться останавливать и запускать службы с помощью systemd.

Решение

Используйте команду `systemctl`. Следующие команды демонстрируют управление службами на примере службы SSH.

Запуск службы:

```
$ sudo systemctl start sshd.service
```

Остановка службы:

```
$ sudo systemctl stop sshd.service
```

Остановка и перезапуск службы:

```
$ sudo systemctl restart sshd.service
```

Перезагрузка конфигурации службы. Например, вы внесли изменения в `sshd_config` и хотите, чтобы эти изменения вступили в силу без перезапуска службы:

```
$ sudo systemctl reload sshd.service
```

Комментарий

Все эти команды могут управлять сразу несколькими службами, для чего достаточно перечислить их через пробел, например:

```
$ sudo systemctl start sshd.service mariadb.service firewalld.service
```

Если вам интересно, какие команды systemd выполняет «за кулисами», чтобы запустить, перезагрузить или остановить отдельный демон, загляните в соответствующие файлы модулей. Некоторые службы содержат инструкции по запуску, перезагрузке, остановке в своих файлах модулей, как в этом примере для httpd:

```
ExecStart=/usr/sbin/httpd/ $OPTIONS -DFOREGROUND
ExecReload=/usr/sbin/httpd $OPTIONS -k graceful
ExecStop=/bin/kill -WINCH ${MAINPID}
```

Вам не нужно делать ничего особенного с этой информацией; она приводится здесь, только чтобы показать, как `systemctl` управляет конкретной службой.

Дополнительная информация

- Рецепт 4.6.
- `man 1 systemctl`

4.6. Включение и выключение служб

Задача

Включить службу или службы, чтобы они автоматически запускались при загрузке системы, либо, напротив, предотвратить запуск службы при загрузке или полностью отключить ее.

Решение

Включение службы настраивает ее на автоматический запуск при загрузке.

Выключение службы предотвращает ее запуск при загрузке, но сохраняет возможность запускать и останавливать ее вручную.

Маскирование выключает службу так, что ее вообще нельзя запустить.

Ниже представлен пример включения службы sshd:

```
$ sudo systemctl enable sshd.service
Created symlink /etc/systemd/system/multi-user.target.wants/sshd.service →
/usr/lib/systemd/system/sshd.service
```

Как показывает вывод, включение службы сводится к созданию в каталоге `/etc/systemd/system/` символической ссылки на файл службы в каталоге `/lib/systemd/system/`. Включение службы не приводит к ее немедленному запуску.

Запустить службу можно, введя команду `systemctl start` или добавив параметр `--now` в команду включения:

```
$ sudo systemctl enable --now sshd.service
```

Следующая команда выключит службу sshd. Она не останавливает службу немедленно, поэтому после выключения ее нужно остановить вручную:

```
$ sudo systemctl disable sshd.service
Removed /etc/systemd/system/multi-user.target.wants/sshd.service
$ sudo systemctl stop sshd.service
```

Кроме того, службу можно выключить и остановить с помощью одной команды:

```
$ sudo systemctl disable --now sshd.service
```

Следующая команда повторно включает службу mariadb — сначала выключает, а затем вновь включает ее. Если вы создали символическую ссылку для службы вручную, то эта команда поможет быстро восстановить значение по умолчанию:

```
$ sudo systemctl reenable mariadb.service
Removed /etc/systemd/system/multi-user.target.wants/mariadb.service.
Removed /etc/systemd/system/mysqld.service.
Removed /etc/systemd/system/mysql.service.
Created symlink /etc/systemd/system/mysql.service →
/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service →
/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service →
/lib/systemd/system/mariadb.service.
```

Следующая команда полностью выключает службу bluetooth, маскируя ее. В результате эту службу нельзя будет запустить:

```
$ sudo systemctl mask bluetooth.service
Created symlink /etc/systemd/system/bluetooth.service → /dev/null.
```

Размаскирование службы bluetooth не включает ее, поэтому ее нужно запускать вручную:

```
$ sudo systemctl unmask bluetooth.service
Removed /etc/systemd/system/bluetooth.service.
$ sudo systemctl start bluetooth.service
```

Комментарий

Включение, выключение, маскирование и размаскирование службы не влияет на ее текущее состояние, если не был задействован параметр `--now`. Его

можно использовать с командами `enable`, `disable` и `mask`, чтобы немедленно запустить или остановить службу, но он не работает с командой размаскирования `unmask`.

См. подраздел «Комментарий» в рецепте 4.3, чтобы узнать больше о том, как systemd использует символические ссылки для управления службами.

Дополнительная информация

- `man 1 systemctl`
- Подраздел «Комментарий» в рецепте 4.3, содержащий информацию о том, как systemd использует символические ссылки для управления службами.

4.7. Остановка неисправных процессов

Задача

Научиться останавливать неисправные процессы. Некая служба может перестать откликаться, выйти из-под контроля или начать запускать новые копии себя, вызывая зависание системы. Обычная команда остановки не работает. Что можно предпринять?

Решение

Остановка процесса называется его завершением. В системах Linux с systemd следует выполнять команду `systemctl kill`. В системах без systemd используйте устаревшую команду `kill`.

Команда `systemctl kill` предпочтительнее, поскольку останавливает все процессы, принадлежащие службе, и не оставляет ни «осиротевших» процессов, ни процессов, которые могут перезапустить службу и продолжить создавать проблемы. Сначала попробуйте выполнить эту команду без параметров, кроме имени службы, а затем проверьте статус:

```
$ sudo systemctl kill mariadb

$ systemctl status mariadb
● mariadb.service - MariaDB 10.1.44 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset:
enabled)
   Active: inactive (dead) since Sun 2020-06-28 19:57:49 PDT; 6s ago
[...]
```

В этом случае служба была остановлена чисто. Если данная команда не дает желаемого результата, то попробуйте ядерный параметр:

```
$ sudo systemctl kill -9 mariadb
```

Устаревшая команда `kill` не распознает имена служб или команд, поэтому ей нужно передавать идентификатор PID нужного процесса:

```
$ sudo kill 1234
```

Если процесс не остановился, то добавьте в команду ядерный параметр:

```
$ sudo kill -9 1234
```

Комментарий

Используйте команду `top` для выявления неуправляемых процессов. Запустите ее без параметров, и она покажет в самом верху процессы, которые потребляют больше всего вычислительных ресурсов. Нажмите клавишу `Q`, чтобы выйти из `top`.

```
$ top
top - 20:30:13 up  4:24,  6 users,  load average: 0.00, 0.03, 0.06
Tasks: 246 total,   1 running, 170 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.4 us,  0.2 sy,  0.0 ni, 99.4 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 16071016 total,  7295284 free,  1911276 used,  6864456 buff/cache
KiB Swap:  8928604 total,  8928604 free,        0 used. 13505600 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU %MEM     TIME+ COMMAND
 3504 madmax    20   0 99.844g 177588  88712 S   2.6  1.1   0:08.68 evolution
 2081 madmax    20   0 3818636 517756 177744 S   0.7  3.2   5:07.56 firefox
 1064 root      20   0  567244 148432 125572 S   0.3  0.9  12:54.75 Xorg
 2362 stash     20   0 2997732 230508 145444 S   0.3  1.4   0:40.72 Web Content
[...]
```

Команда `kill` посылает сигналы процессам и по умолчанию использует сигнал SIGTERM (signal terminate — сигнал завершения). SIGTERM — щадящий сигнал, который позволяет процессу аккуратно завершить свою работу. Однако процессы могут его игнорировать. Сигналы можно указывать по имени или по номеру; многим из нас проще запомнить число, поэтому вместо имени SIGTERM можно передать номер сигнала:

```
$ sudo kill -1 1234
```

Команда `kill -9` посылает сигнал SIGKILL, который вызывает немедленную и безусловную остановку процесса, а также всех его потомков.

Остановить службу с помощью команды `systemctl kill` проще и надежнее, чем с помощью `kill`. Вам нужно только указать название службы и не надо искать ее идентификатор PID. Команда `systemctl kill` гарантирует остановку всех процессов, принадлежащих службе, чего не может гарантировать `kill`.

За долгие годы накопилось множество сигналов, и вы можете прочитать о них в руководстве `man 7 signal`. По моему опыту, чаще других используются сигналы SIGTERM и SIGKILL, но пусть это не останавливает вас от знакомства с другими сигналами.

Возможно, вам, как и мне, не нравится такая терминология, как убийство[1], предки, потомки и сироты. Будем надеяться, что в будущем она изменится.

Дополнительная информация

- `man 5 systemd.kill`
- `man 1 systemctl`
- `man 1 kill`
- `man 7 signal`

4.8. Управление уровнями запуска с помощью systemd

Задача

Выполнить перезагрузку системы с состоянием по выбору, как при использовании уровней запуска SysV.

Решение

Система инициализации systemd поддерживает *цели* (targets), напоминающие уровни запуска (runlevels) в SysV. Это профили загрузки, которые обеспечивают запуск системы с различными настройками, такими как многопользовательский режим с графическим рабочим столом, многопользовательский режим без графического рабочего стола, а также аварийный и восстановительный режимы для использования в случаях, когда система с текущим профилем (целью) не загружается. (См. подраздел «Комментарий» ниже, в котором приводится дополнительная информация об уровнях запуска.)

[1] Команду kill можно перевести как «убить».

Следующая команда проверяет, работает ли система, и сообщает ее состояние:

```
$ systemctl is-system-running
running
```

Сообщает текущую цель по умолчанию:

```
$ systemctl get-default
graphical.target
```

Сообщает текущий уровень запуска:

```
$ runlevel
N 5
```

Перезагрузить систему в режиме восстановления:

```
$ sudo systemctl rescue
```

Перезагрузить систему в аварийном режиме:

```
$ sudo systemctl emergency
```

Перезагрузить систему в режиме по умолчанию:

```
$ sudo systemctl reboot
```

Перезагрузить в другом режиме без изменения режима по умолчанию:

```
$ sudo systemctl isolate multi-user.target
```

Установить уровень запуска по умолчанию:

```
$ sudo systemctl set-default multi-user.target
```

Вывести список имеющихся файлов, определяющих уровни запуска:

```
$ ls -l /lib/systemd/system/runlevel*
```

Вывести список зависимостей для выбранного уровня запуска:

```
$ systemctl list-dependencies graphical.target
```

Комментарий

Уровни запуска в SysV — это разные состояния, в которые может загружаться система, например, с графическим рабочим столом, без графического рабочего стола, а также в аварийном и восстановительном режиме, если система не загружается с уровнем запуска по умолчанию.

Цели в systemd примерно соответствуют устаревшим уровням запуска в SysV:

- `runlevel0.target`, `poweroff.target` — остановка системы;
- `runlevel1.target`, `rescue.target` — однопользовательский режим без графической среды, все локальные файловые системы монтируются, вход может выполнить только пользователь root, сеть неактивна;
- `runlevel3.target`, `multi-user.target` — многопользовательский режим без графической среды;
- `runlevel5.target`, `graphical.target` — многопользовательский режим с графической средой;
- `runlevel6.target`, `reboot.target` — перезагрузка.

Команда `systemctl emergency` — это особая аварийная цель, более ограниченная, чем режим восстановления `rescue`: в этом режиме не запускаются службы, не монтируются файловые системы, кроме корневой, нет сети и вход может выполнить только пользователь root. Это самая минимальная работающая конфигурация системы, предназначенная для устранения проблем. Варианты загрузки в аварийном и восстановительном режимах доступны в меню загрузчика GRUB2.

Команда `systemctl is-system-running` сообщает текущее состояние системы, которое может быть одним из следующих:

- `initializing` — система еще не завершила запуск;
- `starting` — система на заключительном этапе запуска;
- `running` — система полностью работоспособна и все процессы запущены;
- `degraded` — система работоспособна, но один или несколько модулей systemd потерпели неудачу. Выполните `systemctl | grep failed`, чтобы увидеть, какие это модули;
- `maintenance` — система загружена в аварийном (`emergency`) или восстановительном (`rescue`) режиме;
- `stopping` — systemd останавливается;
- `offline` — systemd не запущена;
- `unknown` — существует проблема, не позволяющая systemd определить текущее состояние.

Дополнительная информация

- `man 1 systemctl`
- `man 8 systemd-halt.service`

4.9. Диагностика медленного запуска

Задача

systemd обещает быструю загрузку, но ваша система запускается медленно и хотелось бы узнать почему.

Решение

Выполните команду `systemd-analyze blame` без параметров, чтобы увидеть список системных процессов и время их запуска:

```
$ systemd-analyze blame
         34.590s apt-daily.service
          6.782s NetworkManager-wait-online.service
          6.181s dev-sda2.device
          4.444s systemd-journal-flush.service
          3.609s udisks2.service
          2.450s snapd.service
[...]
```

Чтобы проанализировать только пользовательские процессы:

```
$ systemd-analyze blame --user
          3.991s pulseaudio.service
           553ms at-spi-dbus-bus.service
           380ms evolution-calendar-factory.service
           331ms evolution-addressbook-factory.service
           280ms xfce4-notifyd.service
[...]
```

Комментарий

Часто полезно посмотреть, что же запускается при загрузке, и, может быть, даже найти службы, которые не должны запускаться в этот момент. Я люблю отключать Bluetooth, поскольку не использую его на своих серверах или ПК, но многие дистрибутивы Linux включают его по умолчанию.

Дополнительная информация

- `man 1 systemd-analyze`

ГЛАВА 5

Управление пользователями и группами

В Linux есть два типа пользователей: пользователи-люди и системные пользователи. Каждый из них имеет уникальный идентификатор (User ID, UID) и по крайней мере один идентификатор группы (Group ID, GID). Каждый пользователь входит в одну основную группу и может входить в несколько дополнительных.

Каждый пользователь-человек владеет домашним каталогом со своими личными файлами. Домашние каталоги пользователей находятся в `/home`, и их названия совпадают с именами владельцев, как в нашем примере пользователя Duchess, которому принадлежит каталог `/home/duchess`. Помимо основной группы, пользователи могут входить в несколько других, которые называются *дополнительными*. Пользователи в группе получают все ее привилегии. (Чтобы узнать больше о привилегиях, см. главу 6.) Привилегии управляют доступом к файлам и командам и являются основой безопасности системы.

Системные пользователи — это системные службы и процессы. Учетные записи системных пользователей нужны лишь для управления привилегиями, и у них нет паролей и каталогов в `/home`.

Пользователи-люди делятся на две категории: пользователь root, или суперпользователь, обладает неограниченными привилегиями и может делать в системе все что угодно. Все остальные пользователи называются обычными, или непривилегированными. Обычным дается достаточно прав для управления их файлами и выполнения команд, которые разрешается использовать этой категории людей. Обычным пользователям могут быть предоставлены ограниченные или полные привилегии root, о которых вы узнаете из рецептов, описывающих команды `su` и `sudo`.

Увидеть список всех пользователей в системе можно в файле `/etc/passwd`, а все группы — в `/etc/group`.



Централизованное управление пользователями

Файлы /etc/passwd и /etc/group достались в наследство от Unix и практически не изменились с тех пор, как перекочевали в Linux в 1992 году. С тех пор появились новые инструменты для управления пользователями и группами, например централизованные базы данных, обслуживающие целые организации. В этой главе мы не будем рассматривать инструменты централизованного управления пользователями.

В Linux есть несколько команд для управления пользователями и группами:

- `useradd` — создает новых пользователей;
- `groupadd` — создает новые группы;
- `userdel` — удаляет пользователей;
- `groupdel` — удаляет группы;
- `usermod` — изменяет настройки существующего пользователя;
- `passwd` — создает и изменяет пароли.

Они являются частью набора *Shadow Password Suite*, и основным конфигурационным файлом для них служит `/etc/login.defs`.

Команда `useradd` действует по-разному в разных системах, в зависимости от настроек. Традиционно эта команда объединяла всех новых пользователей в одну и ту же основную группу `users` (100). Это означало, что пользователи должны были проявлять осторожность при выборе разрешений для своих файлов, чтобы случайно не раскрыть свои секреты другим членам группы. В Red Hat изменили данную ситуацию, разработав схему *User Private Group*, согласно которой для каждого нового пользователя создается личная основная группа. Большинство дистрибутивов Linux используют эту схему по умолчанию, хотя есть исключения, такие как openSUSE.

Набор команд Shadow Password Suite был создан Джулианной Фрэнсис Хо (Julianne Frances Haugh) в 1980-х годах, еще до появления Linux, для повышения безопасности паролей Unix и упрощения управления учетными записями пользователей. В 1992-м этот набор был перенесен в систему Linux, когда ей едва исполнился год.

До появления Shadow Password Suite все файлы, имеющие отношение к учетным записям пользователей, приходилось редактировать по отдельности, имелось несколько команд управления паролями, а хешированные пароли хранились в файлах `/etc/passwd` и `/etc/group`. Но, поскольку `/etc/passwd` должен оставаться доступным для чтения всем пользователям, хранение паролей в нем, пусть и в зашифрованном виде, чревато неприятностями. Скопировав этот файл, любой желающий теоретически сможет взломать пароли. Перемещение хешированных паролей в файлы `/etc/shadow` и `/etc/gshadow`, доступные только пользователю

root, дополнило защиту. Долгожительство Shadow Password Suite свидетельствует о том, насколько хорошо был проработан и реализован данный пакет.

Относительно недавно в Debian появились *adduser* и *addgroup*. Это сценарии-обертки на Perl для команд `useradd` и `groupadd`. Они по шагам проведут вас через процесс создания нового пользователя и новой группы.

В этой главе вы узнаете, как создавать и удалять обычных и системных пользователей, управлять паролями, определять идентификаторы UID и GID, устанавливать желаемые значения по умолчанию настроек для создания новых пользователей, изменять принадлежность к группам, настраивать общие файлы для новых пользователей, очищать каталоги после удаления пользователей, получать привилегии root и предоставлять ограниченные полномочия root обычным пользователям.

5.1. Определение UID и GID пользователя

Задача

Определить UID и GID пользователя.

Решение

Выполнив команду `id` без параметров, можно узнать собственные UID и GID. Ниже представлен пример определения идентификаторов пользователя Duchess:

```
duchess@pc:~$ id
uid=1000(duchess) gid=1000(duchess)
groups=1000(duchess),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),118(lpadmin),
126(sambashare),131(libvirt)
```

Чтобы узнать UID и GID другого пользователя, нужно передать команде `id` его имя:

```
duchess@pc:~$ id madmax
uid=1001(madmax) gid=1001(madmax) groups=1001(madmax),1010(composers)
```

С помощью `id` можно также узнать свой эффективный идентификатор пользователя, когда вы выступаете от имени другого пользователя. Ниже представлен пример, в котором задействована с команда `sudo`:

```
duchess@client4:~$ sudo id -un
root

duchess@client4:~$ sudo -u madmax id -gn
madmax
```

Комментарий

В Linux есть три типа идентификаторов пользователя/группы:

- реальный UID/GID;
- эффективный UID/GID;
- сохраненный UID/GID.

Реальный идентификатор — это UID и GID основной группы, присвоенные пользователю при создании. Это то, что вы видите, когда запускаете команду `id` без параметров от своего имени.

Эффективный идентификатор — это UID, используемый для запуска процесса, которому требуются привилегии, отличные от привилегий пользователя, запускающего процесс. Примером может служить команда `passwd`, которая требует привилегий суперпользователя и применяет специальные режимы разрешений, чтобы дать пользователям возможность изменять свои собственные пароли.

Вы можете убедиться в этом сами. Во-первых, взгляните на разрешения команды `passwd`:

```
$ ls -l /usr/bin/passwd
-rwsr-xr-x 1 root root 68208 May 27 2020 /usr/bin/passwd
```

Как видите, файл `passwd` принадлежит пользователю и группе root. Теперь введите команду `passwd` и нажмите Enter.

Откройте второй терминал, найдите идентификатор процесса `passwd`, а затем по этому идентификатору определите эффективный и реальный идентификатор пользователя и эффективный идентификатор группы:

```
$ ps -a|grep passwd
12916 pts/1 00:00:00 passwd

$ ps -eo pid,euser,ruser,rgroup | grep 12916
12916 root root root
```

Несмотря на то что команда `passwd` была запущена непривилегированным пользователем, она работает с правами root. (См. рецепт 6.11, в котором рассказывается о режимах специальных разрешений.)

Сохраненный идентификатор используется процессами, которым требуются повышенные привилегии, обычно привилегии root. Когда для выполнения работы достаточно привилегий обычного пользователя, он может временно переключиться на непривилегированный идентификатор пользователя. Эффективным становится UID с пониженными привилегиями, а исходный эффективный UID

сохраняется в SUID — Saved User ID (сохраненный идентификатор пользователя). Когда процессу снова требуются повышенные привилегии, он назначает эффективным сохраненный SUID.

Команда `id` поддерживает несколько параметров:

- `-u` выводит эффективный числовой UID;
- `-g` выводит эффективный числовой GID;
- `-G` выводит все числовые идентификаторы групп;
- `-n` выводит имя пользователя вместо числового идентификатора UID. Этот параметр можно использовать в комбинации с `-u`, `-g` и `-G`;
- `-un` выводит эффективный числовой UID и имя пользователя;
- `-gn` выводит имя эффективной группы;
- `-Gn` выводит все имена эффективных групп;
- `-r` выводит реальный числовой идентификатор вместо эффективного. Этот параметр можно использовать в комбинации с `-u`, `-g` и `-G`.

Дополнительная информация

- Рецепт 6.11.
- `man 1 id`
- `man 1 ps`

5.2. Создание учетной записи для пользователя-человека с помощью команды useradd

Задача

Создать новую учетную запись пользователя с личной группой и домашний каталог с набором файлов по умолчанию, таких как `.bashrc`, `.profile`, `.bash_history`, и любыми другими необходимыми файлами.

Решение

В большинстве дистрибутивов Linux для этой цели имеется команда `useradd`, которую можно настроить под свои требования. Конфигурация по умолчанию

различается в разных дистрибутивах, поэтому самый простой способ узнать используемые настройки — создать новую пробную учетную запись:

```
$ sudo useradd test1
```

После этого запустите команду `id` и проверьте — был ли создан домашний каталог. Следующий пример был получен в Fedora 34:

```
$ id test1
uid=1011(test1) gid=1011(test1) groups=1011(test1)

$ sudo ls -a /home/test1/
.  ..  .bash_logout  .bash_profile  .bashrc
```

В этом примере конфигурация по умолчанию соответствует требованиям, перечисленным выше в подразделе «Задача». Теперь осталось только установить пароль:

```
$ sudo passwd test1
Changing password for user test1.
New password: пароль
Retype new password: пароль
passwd: all authentication tokens updated successfully.
```

При необходимости можно заставить пользователя сменить пароль при первом входе в систему после того, как вы создали пароль:

```
$ sudo passwd -e test1
Expiring password for user test1.
passwd: Success
```

Сообщите данные для входа вашему пользователю, и он начнет использовать свою учетную запись. Новая учетная запись в файле `/etc/passwd` представлена следующим образом:

```
test1:x:1011:1011::/home/test1:/bin/bash
```

В некоторых дистрибутивах, например в openSUSE, команда `useradd` настроена так, что по умолчанию не создает домашний каталог пользователя и включает всех пользователей в группу `users (100)`. Вследствие этого другие пользователи смогут получить доступ к файлам друг друга, если разрешения файлов для группы позволяют это. Следующий пример создает личную группу пользователя:

```
$ sudo useradd -mU test2
```

Параметр `-m` позволит создать домашний каталог пользователя, а параметр `-U` — личную группу с именем, совпадающим с именем пользователя.

Комментарий

Все новые учетные записи пользователей остаются неактивными до установки пароля.

Первая группа, в которую добавляется пользователь, будь то его личная группа или общая группа для всех пользователей, становится его *основной* группой. Все остальные группы, в которые включается пользователь, считаются *дополнительными*.

Ниже представлено еще несколько полезных параметров:

- `-G`, `--groups` — добавляет пользователя в несколько дополнительных групп, перечисленных через запятую. Группы должны существовать к моменту выполнения команды:

  ```
  $ sudo useradd -G group1,group2,group3 test1
  ```

- `-c`, `--comment` — принимает любую текстовую строку и сохраняет ее как полное имя пользователя, комментарий или описание:

  ```
  $ useradd -G group1,group2,group3 -c 'Test 1,,,,' test1
  ```

Четыре запятых в данном примере определяют пять полей: имя, номер кабинета, рабочий телефон, домашний телефон и прочее (произвольная информация). В прошлом эти поля назывались данными GECOS, где GECOS (General Electric Comprehensive Operating Supervisor) — название операционной системы для мейнфрейма. Вы можете ввести в эти поля любую информацию по своему усмотрению или оставить их пустыми, хотя иногда имеет смысл указать полное имя пользователя. Изучите свой файл `/etc/passwd` и посмотрите, как другие учетные записи задействуют поля GECOS.

Настройки по умолчанию для команды `useradd` разбросаны по нескольким конфигурационным файлам; см. рецепт 5.4, чтобы узнать, как их изменить.

Дополнительная информация

- `man 8 useradd`
- `man 5 login.defs`
- `/etc/default/useradd`
- `/etc/skel`
- `/etc/login/defs`

5.3. Создание системной учетной записи с помощью команды useradd

Задача

Создать системного пользователя с помощью команды `useradd`.

Решение

Следующий пример создаст нового системного пользователя без домашнего каталога, без оболочки входа и с UID из диапазона, предназначенного для системных пользователей:

```
$ sudo useradd -rs /bin/false service1
```

Параметр `-r` создает системного пользователя с реальным UID из диапазона, предназначенного для системных пользователей, а параметр `-s` задает оболочку входа `/bin/false` — команду, которая ничего не делает и не позволяет выполнить вход в систему с именем этого пользователя.

Дополнительную информацию о диапазонах UID и GID вы найдете в подразделе «Комментарий» рецепта 5.6.

Комментарий

Раньше большинство служб выполнялись с привилегиями пользователя nobody. В настоящее время общепринято создавать для служб своих отдельных пользователей, так как это обеспечивает более высокий уровень безопасности, чем применение одного пользователя nobody, владеющего несколькими службами. Вам редко придется создавать учетные записи системных пользователей, поскольку службы делают это автоматически при установке.

Пользователь nobody всегда получает UID 65534 и GID 65534.

Дополнительная информация

- `man 8 useradd`
- `man 1 false`
- Подраздел «Комментарий» в рецепте 5.6.

5.4. Изменение настроек по умолчанию для команды useradd

Задача

Настройки по умолчанию для команды `useradd` вам не подходят, и их нужно изменить.

Решение

Настройки команды `useradd` разбросаны по множеству конфигурационных файлов, таких как `/etc/default/useradd`, `/etc/login.defs` и файлы в каталоге `/etc/skel`.

В файле `/etc/default/useradd` находятся следующие настройки. Этот пример взят из openSUSE:

```
$ useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

`GROUP=100` назначает единую группу с идентификатором 100 как основную всем новым пользователям. Чтобы обеспечить включение всех новых пользователей в общую группу, нужно создать эту группу, выключить параметр `USERGROUPS_ENAB` в `/etc/login.defs` и присвоить GID общей группы параметру `GROUP=` в `/etc/default/useradd`. Если, например, допустить, что наш пользователь Duchess включен в общую группу, то команда `id` выведет для него `uid=1000(duchess) gid=100(users)`.

Чтобы обеспечить создание личных групп для всех новых пользователей, нужно присвоить параметру `USERGROUPS_ENAB` в `/etc/login.defs` значение `yes` и закомментировать параметр `GROUP=` в `/etc/default/useradd`. Если, например, допустить, что наш пользователь Duchess имеет личную группу, то команда `id` выведет для него `uid=1000(duchess) gid=1000(duchess)`.

- `HOME=` задает каталог по умолчанию для размещения домашних каталогов пользователей. По умолчанию `/home`.

- `INACTIVE=-1` задает срок действия пароля в днях, по истечении которого учетная запись будет заблокирована. Значение `0` сразу же отключает учетную запись, так как срок действия пароля истекает немедленно, а значение `-1` запрещает блокирование учетных записей.
- `EXPIRE=` назначает конечную дату действия учетной записи в формате YYYY-MM-DD. Например, если установить значение `2021-12-31`, то учетная запись будет заблокирована в эту дату. Если параметр `EXPIRE=` оставить пустым, то это будет означать отсутствие конечной даты.
- `SHELL=/bin/bash` назначает командную оболочку по умолчанию. Наиболее широко используется оболочка `/bin/bash`. Вдобавок в этом параметре можно назначить любую другую командную оболочку, установленную в системе, например: `/bin/zsh` или `/usr/bin/tcsh`. Получить список установленных командных оболочек можно с помощью команды `cat /etc/shells`.
- `SKEL=/etc/skel` определяет каталог с файлами, которые должны автоматически копироваться в домашние каталоги новых пользователей. В большинстве дистрибутивов Linux такие файлы помещаются в `/etc/skel`. К ним относятся: `.bash_logout`, `.bash_profile`, `.profile`, `.bashrc` и любые другие файлы, которые должны иметься у новых пользователей. Вы можете отредактировать эти файлы в соответствии со своими требованиями. `SKEL` — это сокращение от skeleton («каркас, основа»).
- `CREATE_MAIL_SPOOL=yes` — пережиток прошлого, и этому параметру всегда следует присваивать значение `yes`, так как некоторые устаревшие процессы могут все еще нуждаться в нем.

Ниже представлены наиболее актуальные параметры со значениями по умолчанию в `/etc/login.defs`:

- `USERGROUPS_ENAB yes` включает создание личной группы для каждого пользователя;
- `CREATE_HOME yes` требует от `useradd` автоматически создавать домашние каталоги для новых пользователей. Не применяется к системным пользователям (см. рецепт 5.3).

Комментарий

Диапазоны UID определены в `/etc/login.defs`. Каждый UID должен быть уникальным, поэтому команды создания учетных записей пользователей назначают UID из диапазона, определенного в данном файле. Обычно диапазон UID для учетных записей людей начинается с 1000 и автоматически используется

командой `useradd`. С помощью параметра `-u` можно назначить конкретный UID, но при этом он не должен использоваться никакой другой учетной записью и соответствовать настроенной схеме нумерации (см. подраздел «Комментарий» в рецепте 5.6).

Обязательная смена пароля при первом входе в систему — это простая мера предосторожности против утечки исходного пароля в чужие руки при передаче от администратора к пользователю.

Дополнительная информация

- `man 8 useradd`
- `man 5 login.defs`
- `/etc/default/useradd`
- `/etc/skel`
- `/etc/login/defs`

5.5. Настройка каталогов для документов, музыки, видео, изображений и загрузок

Задача

Организовать создание новых пользователей в соответствии с рецептом и настройку каталогов для документов, музыки, видео, изображений и загрузок.

Решение

Создание этих каталогов является функцией не команды `useradd`, а, скорее, инструмента управления пользовательскими каталогами X Desktop Group (XDG). Отдельные каталоги для документов, музыки, видео считаются специализированными, и соответствующие им настройки, применяемые по умолчанию ко всем пользователям, находятся в конфигурационном файле `/etc/xdg/user-dirs.defaults`:

```
$ less /etc/xdg/user-dirs.defaults
# Настройки пользовательских каталогов по умолчанию
```

```
#
# Значения — это пути относительно домашнего каталога,
# и названия каталогов будут переведены поэлементно на язык пользователя
# в соответствии с его региональными настройками
DESKTOP=Desktop
DOWNLOAD=Downloads
TEMPLATES=Templates
PUBLICSHARE=Public
DOCUMENTS=Documents
MUSIC=Music
PICTURES=Pictures
VIDEOS=Videos
# Другие возможные варианты:
#MUSIC=Documents/Music
#PICTURES=Documents/Pictures
#VIDEOS=Documents/Videos
```

Настройки представлены парами «имя — значение». Имена нельзя изменить. Значения — это каталоги, соответствующие именам, находящиеся в домашних каталогах пользователей. Например, имя `DOCUMENTS` отображается в каталог `/home/username/Documents`. Каталоги создаются автоматически для каждого нового пользователя при первом запуске графической среды рабочего стола. Вы можете закомментировать любые каталоги, чтобы исключить их создание, или изменить названия каталогов.

Пользователи могут определять свои настройки в `~/.config/user-dirs.dirs`. Каталоги должны существовать до применения изменений. Ниже представлен пример для пользователя Duchess, которому не нравятся значения по умолчанию. Обратите внимание, что синтаксис «имя — значение» в `~/.config/user-dirs.dirs` отличается:

```
XDG_DESKTOP_DIR="$HOME/table"
XDG_DOWNLOAD_DIR="$HOME/landing-zone"
XDG_DOCUMENTS_DIR="$HOME/omg-paperwork"
XDG_MUSIC_DIR="$HOME/singendance"
XDG_PICTURES_DIR="$HOME/piccies"
```

Внеся изменения и создав нужные каталоги, выполните команду `xdg-user-dirs-update`, чтобы применить изменения:

```
duchess@pc:~$ xdg-user-dirs-update --set DOWNLOAD $HOME/landing-zone
duchess@pc:~$ xdg-user-dirs-update --set DESKTOP $HOME/table
duchess@pc:~$ xdg-user-dirs-update --set DOCUMENTS $HOME/omg-paperwork
duchess@pc:~$ xdg-user-dirs-update --set MUSIC $HOME/singendance
duchess@pc:~$ xdg-user-dirs-update --set PICTURES $HOME/piccies
```

Выйдите из системы, затем войдите снова, и вы должны увидеть то, что показано на рис. 5.1. Инструмент XDG применил соответствующие значки к специализированным каталогам.

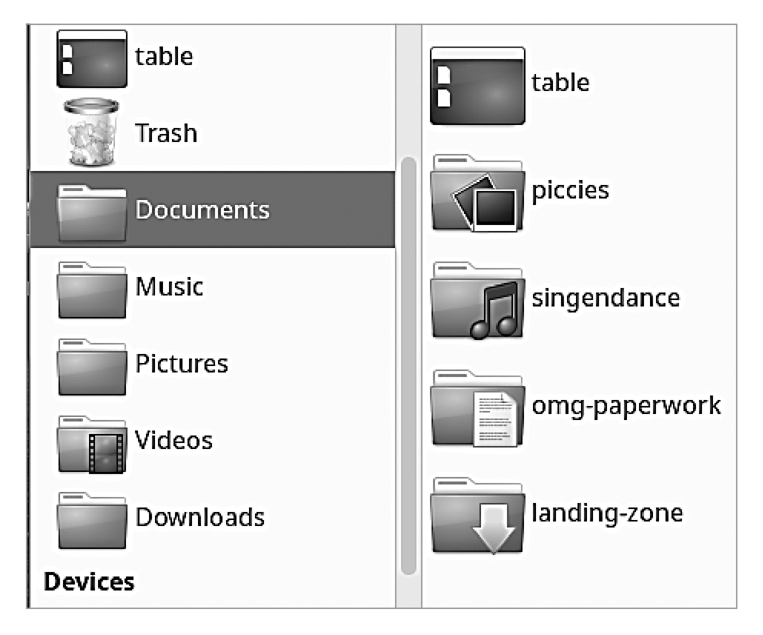


Рис. 5.1. Специализированные каталоги после изменения настроек

Ярлыки на боковой панели не изменятся, как и старые каталоги, но на них уже не будут отображаться специальные значки. Вам придется вручную изменить ярлыки и перенести содержимое из старых каталогов в новые.

Восстановить настройки по умолчанию из `/etc/xdg/user-dirs.defaults` можно с помощью команды:

```
$ xdg-user-dirs-update --force
```

Выйдите и войдите снова, чтобы убедиться, что ни один из ваших каталогов не был удален и не претерпел никаких изменений, кроме отображения специальных значков в диспетчере файлов.

Комментарий

В команде `xdg-user-dirs-update --set` можно использовать только имена, перечисленные в руководстве `man 5 user-dirs.default`:

```
DESKTOP
DOWNLOAD
TEMPLATES
PUBLICSHARE
DOCUMENTS
```

```
MUSIC
PICTURES
VIDEOS
```

Значениями могут быть только целевые каталоги. Пути к целевым каталогам откладываются относительно домашних каталогов пользователей. Чтобы использовать каталоги за пределами домашнего каталога, создайте символические ссылки. Например, пользователь Duchess может владеть каталогом `/users/stuff/duchess` и хранить в нем музыкальные файлы. Следующая команда создаст символическую ссылку `/home/duchess/singendance`, указывающую на этот каталог:

```
duchess@pc:~$ ln -s /users/stuff/duchess /home/duchess/singendance
```

Дополнительная информация

- `man 5 user-dirs.defaults`
- `man 1 xdg-user-dirs-update`
- `man 5 user-dirs.conf`
- Описание `xdg-user-dirs` на сайте freedesktop (https://oreil.ly/FFDga).

5.6. Создание пользовательских и системных групп с помощью команды groupadd

Задача

Создать группу с помощью команды `groupadd`.

Решение

Следующий пример создаст новую группу пользователей `musicians`:

```
$ sudo groupadd musicians
```

Чтобы создать системную группу, команду `groupadd` следует вызвать с параметром `-r`:

```
$ sudo groupadd -r service1
```

Комментарий

Системные группы отличаются от групп пользователей-людей диапазонами GID. Они определены в `/etc/login.defs` и используются командами `groupadd` и `useradd`, как показано в следующем примере, полученном в Fedora 34:

```
# Мин./макс. значения для автоматического выбора uid в useradd(8)
#
UID_MIN 1000
UID_MAX 60000
# Системные пользователи
SYS_UID_MIN 201
SYS_UID_MAX 999
# Диапазон вторичных uid для каждого пользователя
SUB_UID_MIN 100000
SUB_UID_MAX 600100000
SUB_UID_COUNT 65536
#
# Мин./макс. значения для автоматического выбора gid в groupadd(8)
#
GID_MIN 1000
GID_MAX 60000
# Системные пользователи
SYS_GID_MIN 201
SYS_GID_MAX 999
# Диапазон вторичных gid для каждого пользователя
SUB_GID_MIN 100000
SUB_GID_MAX 600100000
SUB_GID_COUNT 65536
```

Эти настройки определяют диапазоны идентификаторов, доступных системному администратору. Все остальные диапазоны зарезервированы и управляются системой.

GID автоматически выбираются с помощью команды `groupadd` в соответствии с диапазонами в `/etc/login.defs`. Задать свое значение GID позволяет параметр `-g`, при этом выбранный вами GID должен попадать в соответствующий диапазон и еще не использоваться.

Дополнительная информация

- `man 8 groupadd`
- `/etc/login.defs`

5.7. Добавление пользователей в группы с помощью команды usermod

Задача

Включить пользователя в группу.

Решение

Использовать команду `usermod`. Ниже представлен пример добавления пользователя Duchess в группу `musicians`:

```
$ sudo usermod -aG musicians duchess
```

Следующая команда добавит Duchess в несколько групп:

```
$ sudo usermod -aG musicians,composers,stagehands duchess
```

Альтернативное решение — отредактировать файл `/etc/group` и добавить имя пользователя Duchess во все группы, в которые он должен входить. Члены группы должны перечисляться через запятую, без пробелов, например:

```
musicians:x:900:stash,madmax,duchess
```



Будьте внимательны: добавляйте в конец, но не заменяйте

Если вы забудете параметр -a и укажете только -G, то все пользователи, прежде входившие в группу, будут исключены из нее. Это особенно чревато неприятностями при удалении пользователей из их группы sudo.

После того как членство в группах пользователей, вошедших в систему, будет изменено, они должны выйти из системы и снова войти, чтобы изменения вступили в силу. Существуют разные обходные пути активации членства в группах без выхода из системы, но все они имеют ограничения, например обусловленные особенностями текущей командной оболочки. Группы перечисляются при входе в систему, поэтому наиболее надежным решением является выход и повторный вход.

Комментарий

Параметр `-a` означает append («добавить в конец»), а `-G` — group («группа»).

Дополнительная информация

- `man 8 usermod`

5.8. Создание пользователей с помощью команды adduser в Ubuntu

Задача

Вы используете Debian или другой дистрибутив Linux на основе Debian, и вам хотелось бы узнать, как создать нового пользователя с помощью команды `adduser`.

Решение

Команда `adduser` по шагам проведет вас через процесс создания нового пользователя. Ниже представлен пример создания учетной записи для моего кота Хороняки (Stash):

```
$ sudo adduser stash
Adding user 'stash' ...
Adding new group 'stash' (1009) ...
Adding new user 'stash' (1009) with group 'stash' ...
Creating home directory '/home/stash' ...
Copying files from '/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for stash
Enter the new value, or press ENTER for the default
        Full Name []: Stash Cat
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n]
```

Учетная запись в `/etc/passwd` выглядит следующим образом:

```
stash:x:1009:1009:Stash Cat,,,:/home/stash:/bin/bash
```

Комментарий

Настройки по умолчанию для `adduser` определены в файле `/etc/adduser.conf`, в том числе:

- `DSHELL=` — оболочка входа по умолчанию. Чаще других используется оболочка `/bin/bash`. Кроме того, можно указать любую другую оболочку, установленную в системе, такую как `/bin/zsh` или `/usr/bin/tcsh`. Получить список установленных командных оболочек можно с помощью команды `cat /etc/shells`;
- `USERGROUPS=yes` создает личную группу для каждого нового пользователя;
- параметр `USERS_GID=100` необходим, только если выбрана настройка `USERGROUPS=no`;
- `EXTRA_GROUPS=` — это список дополнительных групп для новых пользователей, например: `EXTRA_GROUPS="audio video plugdev libvirt"`;
- `ADD_EXTRA_GROUPS=1` добавляет по умолчанию новых пользователей в группы, перечисленные в `EXTRA_GROUPS=`.

В `/etc/adduser.conf` определена следующая схема нумерация пользователей и групп:

```
FIRST_SYSTEM_UID=100
LAST_SYSTEM_UID=999

FIRST_SYSTEM_GID=100
LAST_SYSTEM_GID=999

FIRST_UID=1000
LAST_UID=59999

FIRST_GID=1000
LAST_GID=59999--
```

Fedora Linux тоже имеет команду `adduser`, но не настоящий сценарий `adduser`, а всего лишь символическую ссылку на `useradd`:

```
$ stat /usr/sbin/adduser
  File: /usr/sbin/adduser -> useradd
  Size: 7 Blocks: 0 IO Block: 4096 symbolic link
[...]
```

Дополнительная информация

- `man 5 adduser.conf`

5.9. Создание системного пользователя с помощью команды adduser в Ubuntu

Задача

Создать системного пользователя можно с помощью команды `adduser` в Ubuntu (а также в Debian, Mint или другом дистрибутиве, производном от Debian).

Решение

Следующий пример создаст системного пользователя `service1` без домашнего каталога и с собственной основной личной группой:

```
$ sudo adduser --system --no-create-home --group service
Adding system user 'service1' (UID 124) ...
Adding new group 'service1' (GID 135) ...
Adding new user 'service1' (UID 124) with group 'service1' ...
Not creating home directory '/home/service1'.
```

Соответствующая запись в `/etc/passwd` выглядит следующим образом:

```
service1:x:124:135::/home/service1:/usr/sbin/nologin
```

Комментарий

Системные пользователи не имеют домашних каталогов.

Раньше большинство служб выполнялись с привилегиями пользователя nobody, за исключением дистрибутива Debian, в котором использовались пользователь nobody и группа nogroup. Использование одного и того же пользователя для нескольких служб ослабляет безопасность. Вам едва ли придется создавать учетные записи системных пользователей, поскольку диспетчер пакетов обычно создает уникального пользователя и группу при установке новой службы. Но теперь вы знаете, как это сделать, если вдруг понадобится.

Пользователь nobody и группа nogroup всегда получают реальный идентификатор 65534.

Дополнительная информация

- `man 8 adduser`

5.10. Создание пользовательских и системных групп с помощью команды addgroup

Задача

Научиться создавать пользовательские и системные группы с помощью команды `addgroup` в Debian.

Решение

Ниже представлен пример создания группы для пользователей-людей:

```
$ sudo addgroup composers
Adding group 'composers' (GID 1010) ...
Done.
```

Запись для этой группы в `/etc/group` выглядит так:

```
composers:x:1010:
```

Следующий пример создаст системную группу:

```
$ sudo addgroup --system service1
Adding group 'service1' (GID 136) ...
Done.
```

Комментарий

Разница между пользовательскими и системными группами заключается в принадлежности идентификаторов разным диапазонам согласно настройкам в `/etc/adduser.conf`.

Дополнительная информация

- `man 8 addgroup`

5.11. Проверка целостности файла паролей

Задача

От файлов с учетными записями пользователей и групп зависит очень многое, и хотелось бы иметь возможность проверить их целостность.

Решение

Команда pwck проверяет целостность /etc/passwd и /etc/shadow, а grpck проверяет /etc/group и /etc/gshadow. Они проверяют форматирование, допустимость данных и имен, а также действительность GID (полный список см. на страницах справочного руководства man). При запуске без параметров обе команды сообщают как об ошибках, так и о предупреждениях:

```
$ sudo pwck
user 'news': directory '/var/spool/news' does not exist
user 'uucp': directory '/var/spool/uucp' does not exist
user 'www-data': directory '/var/www' does not exist
user 'list': directory '/var/list' does not exist

$ sudo grpck
group mail has an entry in /etc/gshadow, but its password field in /etc/group is
not set to 'x'
grpck: no changes
```

Добавьте параметр -q, чтобы вывести только ошибки:

```
$ sudo pwck -q

$ sudo grpck -q
group mail has an entry in /etc/gshadow, but its password field in /etc/group
is not set to 'x'
```

В данном случае обнаружена ошибка в /etc/gshadow. Это малополезное сообщение, поскольку в действительности не является ошибкой. Пароли редко задаются для групп пользователей, поэтому сообщение об отсутствии пароля как об ошибке только сбивает с толку. Однако другие проверки, например правильное количество полей и уникальное допустимое имя группы, могут быть очень полезными.

Вы никогда не должны редактировать /etc/shadow и /etc/gshadow, только /etc/passwd и /etc/group.

Комментарий

Следующий пример демонстрирует ошибку, которую нужно исправить. Введите **n** в ответ на запрос, чтобы предотвратить удаление записей. Первый запрос delete line (удалить строку) в примере относится к /etc/passwd, а второй — к /etc/shadow:

```
$ sudo pwck -q
invalid password file entry
delete line 'fakeservice:x:996:996::/home/fakeservice'? n
delete line 'fakeservice:!:18469::::::'? n
pwck: no changes
```

Затем исправьте запись в /etc/passwd, и это устранит оба сообщения об ошибках. В данном примере в записи fakeservice:x:996:996::/home/fakeservice отсутствует последнее поле, в действительности запись должна иметь вид: fakeservice:x:996:996::/home/fakeservice:/bin/false.

Предупреждения directory does not exist (каталог не существует) для /etc/passwd обычно относятся к учетным записям системных пользователей, которые не задействуются. Например:

```
user 'www-data': directory '/var/www' does not exist
```

Пользователь www-data не задействуется в отсутствие HTTP-сервера, и каталог /var/www отсутствует, если этот сервер не установлен.

Сообщение no changes (без изменений) означает, что никаких изменений в файле паролей не было произведено.

Полный список проверок вы найдете в странице справочного руководства man.

Дополнительная информация

- man 8 pwck
- man 8 grpck

5.12. Отключение учетной записи пользователя

Задача

Отключить учетную запись пользователя, не удаляя ее.

Решение

Временно деактивировать учетную запись можно, выключив пароль с помощью команды `passwd`:

```
$ sudo passwd -l stash
passwd: password expiry information changed.
```

Теперь пользователь не сможет войти. Следующая команда разблокирует учетную запись:

```
$ sudo passwd -u stash
passwd: password expiry information changed.
```

Однако этот способ не помешает пользователю аутентифицироваться иным способом, например с помощью ключа SSH. Для полного отключения учетной записи примените команду `usermod`:

```
$ sudo usermod --expiredate 1 stash
```

Когда пользователь попытается войти, он увидит сообщение Your account has expired; please contact your system administrator (Срок действия вашей учетной записи истек, обратитесь к своему системному администратору). Чтобы восстановить учетную запись, выполните команду:

```
$ sudo usermod --expiredate -1 stash
```

Комментарий

Еще один способ отключить учетную запись — заменить `x` в поле пароля в файле `/etc/passwd` звездочкой (*):

```
stash:*:1009:1009:Stash Cat,,,:/home/stash:/bin/bash
```

Чтобы восстановить учетную запись, достаточно заменить звездочку символом `x`.

Дополнительная информация

- `man 1 passwd`

5.13. Удаление пользователя с помощью команды userdel

Задача

Удалить пользователя и, возможно, его домашний каталог со всем содержимым.

Решение

Ниже представлен пример, как с помощью команды `userdel` удалить пользователя Stash из `/etc/passwd`, основной группы и из дополнительных групп в теневых файлах:

```
$ sudo userdel stash
```

Если пользователь Stash входит в основную группу, включающую других пользователей (как обсуждалось в рецепте 5.4), то эта группа не удаляется.

Добавьте в команду параметр `-r`, чтобы удалить домашний каталог пользователя с его содержимым и папку электронной почты:

```
$ sudo userdel -r stash
```

Если пользователь владеет файлами за пределами домашнего каталога, то их нужно найти и удалить отдельно (см. рецепт 5.16).

Комментарий

Удалив пользователя, загляните в файлы `/etc/passwd` и `/etc/group` и убедитесь, что все упоминания о пользователе были удалены из них.

Всегда желательно проводить уборку после удаления пользователя.

Дополнительная информация

- `man userdel`

5.14. Удаление пользователя с помощью команды deluser в Ubuntu

Задача

Удалить пользователя с помощью команды `deluser` в Ubuntu (или в другом дистрибутиве, производном от Debian).

Решение

Ниже представлен пример удаления пользователя Stash из `/etc/passwd`, а также из основной группы в `/etc/group` и из дополнительных групп в теневых файлах:

```
$ sudo deluser stash
Removing user 'stash' ...
Warning: group 'stash' has no more members.
Done.
```

Команда `deluser` не удаляет основную группу, если в нее включены другие пользователи, поэтому в случае принадлежности пользователя Stash общей основной группе он не будет удален.

Ниже представлен пример удаления домашнего каталога пользователя Stash и создания резервной копии всех удаляемых файлов:

```
$ sudo deluser --remove-all-files --backup stash
```

Комментарий

Параметр `--backup` создаст в текущем каталоге сжатый архив с файлами пользователя. С помощью параметра `--backup-to` можно указать другой каталог:

```
$ sudo deluser --remove-all-files --backup-to /user-backups stash
```

Если пользователь владеет файлами за пределами домашнего каталога, то их нужно найти и удалить отдельно (см. рецепт 5.16).

Дополнительная информация

- `man 8 deluser`

5.15. Удаление группы с помощью команды delgroup в Ubuntu

Задача

Удалить группу с помощью команды `delgroup` в Ubuntu.

Решение

Следующий пример удалит группу `musicians`:

```
$ sudo delgroup musicians
```

Команда `delgroup` не удаляет основную группу, если в нее включен существующий пользователь, но удалит дополнительные группы несмотря ни на что. Если нежелательно удалять непустые группы, то добавьте в команду параметр `--only-if-empty`:

```
$ sudo delgroup --only-if-empty musicians
```

Комментарий

Поведение по умолчанию команды `delgroup` определяется настройками в `/etc/deluser.conf` и `/etc/adduser.conf`.

Дополнительная информация

- `man 8 delgroup`

5.16. Поиск всех файлов, принадлежащих пользователю

Задача

Отыскать все файлы, принадлежащие пользователю, который был удален.

Решение

Команда `find` способна отыскать в локальной системе все файлы с определенными значениями UID и GID владельца. Следующий пример демонстрирует поиск в корневой файловой системе всех файлов, которыми владеет пользователь с заданным UID:

```
$ sudo find / -uid 1007
```

Поиск может потребовать некоторого времени. Если нет необходимости искать во всей файловой системе, то круг поиска можно сузить определенными подкаталогами, такими как `/etc`, `/home` или `/var`:

```
$ sudo find /etc -uid 1007
$ sudo find /home -uid 1007
$ sudo find /var -uid 1007
```

Вдобавок поиск можно выполнять по GID, а также по имени пользователя или группы:

```
$ sudo find / -gid 1007
$ sudo find / -name duchess
$ sudo find / -group duchess
```

Но что делать с найденными файлами? Как вариант — сменить владельца, передав другому пользователю и переложив бремя решения на него:

```
$ sudo find /backups -uid 1007 -exec chown -v 1010 {} \;
changed ownership of '/backups/duchess/' from 1007 to 1010
changed ownership of '/backups/duchess/bin' from 1007 to 1010
changed ownership of '/backups/duchess/logs' from 1007 to 1010
```

Можно объединить команды `find` и `cp`, чтобы найти и скопировать файлы в другой каталог:

```
$ sudo find / -uid 1007 -exec cp -v {} /orphans \;
```

Команда `cp -v` будет выводить сообщения по мере движения вперед и копировать только файлы, не сохраняя структуру каталогов. Чтобы сохранить ее, добавьте параметр `-r`:

```
$ sudo find / -uid 1007 -exec cp -rv {} /orphans \;
```

Операция копирования оставляет оригинальные файлы на месте. После копирования оригиналы можно удалить. Для этого можно снова выполнить команду `find` и использовать `rm` для удаления оригиналов файлов:

```
$ sudo find / -uid 1007 -exec rm -v {} \;
```

Эта команда удалит файлы, но не каталоги. Чтобы удалить опустевшие каталоги, добавьте параметр `-r`:

```
$ sudo find / -uid 1007 -exec rm -rv {} \;
```

Еще один способ — использовать команды `find` и `mv` для перемещения файлов в другое местоположение:

```
$ sudo find / -uid 1007 -exec mv {} /orphans \;
```

Если вы видите сообщение No such file or directory (Нет такого файла или каталога), то обычно оно связано с тем, что файл или каталог были перемещены, в чем можно убедиться, проверив каталог, куда они были перемещены.

Поиск файлов, принадлежащих несуществующему пользователю или группе:

```
$ find / -nouser
$ find / -nogroup
```

Комментарий

Будьте осторожны с командами `mv` и `rm`, поскольку отменить их действие нельзя. Если вы допустили ошибку, то ваша лучшая надежда на восстановление — это резервная копия.

Уборка за ушедшими пользователями может быть сложной задачей, поскольку компьютеры позволяют легко создавать столько файлов, сколько может вместить хранилище. Если обнаружится, что поиск занимает слишком много времени, то имейте в виду, что, запустив его, вы можете заняться чем-то другим.

Дополнительная информация

- `man 1 find`
- `man 1 mv`
- `man 1 cp`
- `man 1 rm`

5.17. Использование su для получения привилегий root

Задача

Получить привилегии root для выполнения некоторых административных задач.

Решение

Использовать команду su, когда требуется получить привилегии root для администрирования системы:

```
duchess@pc:~$ su -l
Password:
root@pc:~#
```

Если пароль пользователя root вам неизвестен или у него вообще нет пароля, то см. рецепт 5.21, чтобы узнать, как с помощью sudo назначить пароль root.

Завершив работу, выйдите и вернитесь в свою командную оболочку:

```
root@pc:~# exit
logout
duchess@pc:~$
```

Параметр -l настраивает среду и выполняет переход в домашний каталог пользователя root. Выполнив команду su без параметра -l, вы останетесь со своей средой:

```
duchess@pc:~$ su
Password:
root@pc:/home/duchess~#
```

Комментарий

Вы можете получить привилегии любого пользователя, если знаете его пароль.

Команда su дает вам абсолютную власть над системой, и каждая запускаемая вами команда будет выполняться с привилегиями root. Подумайте о возможности использовать команду sudo (см. рецепт 5.18), которая обеспечивает дополнительную безопасность, например защиту пароля root и регистрацию выполняемых действий в журнале аудита.

Дополнительная информация

- man 1 su

5.18. Получение ограниченных привилегий root с помощью команды sudo

Задача

Делегировать некоторые административные задачи другим пользователям, но ограничить их привилегии так, чтобы они могли решать только конкретные задачи.

Решение

Используйте команду `sudo`. Она безопаснее, чем `su`, поскольку предоставляет ограниченные полномочия root определенным пользователям для определенных задач, регистрирует их действия и кэширует пароль пользователя на ограниченное время (по умолчанию на 15 минут). Через 15 минут пользователь должен снова ввести свой пароль. Продолжительность кэширования настраивается. Команда `sudo` защищает пароль root, поскольку пользователи `sudo` вводят собственные пароли.



Некоторые дистрибутивы Linux, такие как openSUSE, для выполнения sudo по умолчанию запрашивают пароль root. См. рецепт 5.22, чтобы узнать, как изменить такое поведение.

Файл `/etc/sudoers` определяет настройки `sudo` и должен редактироваться с помощью специальной команды `visudo`. Она откроет `/etc/sudoers` в вашем текстовом редакторе по умолчанию, и вы сможете просмотреть и отредактировать настройки. Ниже представлен пример выполнения этой команды пользователем Duchess:

```
duchess@pc:~$ sudo visudo
[sudo] password for duchess:
[...]
# Позволить пользователю root выполнять любые команды
root ALL=(ALL) ALL

# Позволить членам группы sudo выполнять любые команды
%sudo ALL=(ALL) ALL
[...]
```

Строка `%sudo ALL=(ALL) ALL` означает, что любой пользователь, включенный в группу `sudo`, получает полные полномочия `sudo`, как и root. Знак процента

указывает, что `%sudo` — это группа из `/etc/group`, а не группа, настроенная в `/etc/sudoers`.

Допустим, у вас есть младший администратор Stash, в обязанности которого входит установка и удаление программного обеспечения, а также обновление системы. Вы можете создать системную группу для Stash. Или определить настройки в `/etc/sudoers` для Stash, позволяющие выполнять эти задачи. Настройки в следующем примере дают пользователю Stash полномочия `sudo` для выполнения перечисленных команд. Для определения полномочий нужно указать имя пользователя, имя хоста локального компьютера и список разрешенных команд, разделенных запятыми:

```
stash server1 = /bin/rpm, /usr/bin/yum, /usr/bin/dnf
```

Теперь допустим, что вы решили расширить полномочия Stash и доверить ему, например, управление службами. Список разрешенных команд в этом случае получится слишком длинным, поэтому можно создать псевдонимы для наборов команд и использовать в настройках эти псевдонимы. В следующем примере определяется псевдоним `SOFTWARE` для команд управления программным обеспечением и псевдоним `SYSTEMD` для команд управления службами:

```
Cmnd_Alias SOFTWARE = /bin/rpm, /usr/bin/yum, /usr/bin/dnf
Cmnd_Alias SYSTEMD = /usr/bin/systemctl start, /usr/bin/systemctl stop,
/usr/bin/systemctl reload, /usr/bin/systemctl restart, /usr/bin/systemctl
status, /usr/bin/systemctl enable, /usr/bin/systemctl disable,
/usr/bin/systemctl mask, /usr/bin/systemctl unmask
```

Теперь настройки для Stash будут выглядеть так:

```
stash server1 = SOFTWARE, SYSTEMD
```

В файле `/etc/sudoers` также можно определить группы пользователей (не связанные с системными группами в `/etc/group`) и присвоить им необходимые псевдонимы команд:

```
User_Alias JRADMIN = stash, madmax

JRADMIN server1 = SOFTWARE, SYSTEMD
```

Можно определить `Host_Alias`, чтобы дать пользователю `sudo` привилегии на выполнение действий на нескольких машинах:

```
Host_Alias SERVERS = server1, server2, server3
```

и затем использовать его в настройках `JRADMIN`:

```
JRADMIN SERVERS = SOFTWARE, SYSTEMD
```

Комментарий

Если пользователь sudo с ограниченными привилегиями попытается выполнить запрещенную команду, он увидит сообщение: Sorry, user duchess is not allowed to execute /some/command as root on server2 (Извините, пользователю duchess не разрешено выполнять /some/command от имени пользователя root на server2).

Не слишком верьте в ограничение возможностей пользователей с помощью определенного набора команд. Многие повседневные приложения поддерживают возможность повышения привилегий через экранирование командной оболочки, и ваши пользователи могут получить полные полномочия root. Ниже представлен пример, как это работает с `awk`:

```
$ sudo awk 'BEGIN {system("/bin/bash")}'
root@client4:/home/duchess#
```

Теперь пользователь Duchess имеет все полномочия root. Ничем не примечательная команда `less` тоже обеспечивает экранирование оболочки. Прочтите файл, используя `less`, который достаточно велик, чтобы потребовалась разбивка на страницы:

```
$ sudo less /etc/systctl.conf
#
# /etc/sysctl.conf — конфигурационный файл для настройки системных переменных
# См. /etc/sysctl.d/ с дополнительными системными переменными.
# См. sysctl.conf (5) для получения дополнительной информации.
/etc/sysctl.conf
```

Введите ***!sh***, затем, когда изменится приглашение к вводу, введите **`whoami`**:

```
duchess@client4:~$ sudo less /etc/systctl.conf
#
# /etc/sysctl.conf — конфигурационный файл для настройки системных переменных
# См. /etc/sysctl.d/ с дополнительными системными переменными.
# См. sysctl.conf (5) для получения дополнительной информации.
!'sh'
duchess@client4:~$ sudo less /etc/sysctl.conf
# whoami
root
```

Введите **`exit`**, чтобы вернуться в обычную оболочку.

По своему опыту могу сказать, что чрезвычайно сложно отследить множество приложений, которые могут обеспечить выход из оболочки. Контролировать ваших пользователей sudo вам поможет `journalctl` (см. рецепт 20.1).

В некоторых дистрибутивах Linux, таких как Fedora, группа `wheel` по умолчанию является группой `sudo`. Загляните в свой файл `/etc/sudoers`, чтобы узнать, какая группа настроена в вашем дистрибутиве. Вы можете также создать свою группу пользователей `sudo` и дать ей любое имя.

Файл `/etc/sudoers` управляет привилегиями только на локальном компьютере. Включение других машин, как в примере с псевдонимом `SERVERS`, позволяет использовать один файл конфигурации на нескольких машинах. Группа `sudo` игнорирует любые элементы, такие как хосты или пользователи, которых нет на локальном компьютере.

Рассмотрим строку `root ALL=(ALL) ALL` и разберемся, что означают все эти `ALL`:

- `root` — поле пользователя, хранящее имя одного пользователя, псевдоним или системную группу;
- `ALL=` — поле хоста. `ALL` в этом поле означает «любой хост», также в этом поле можно указать псевдоним хостов или имя одного хоста;
- `(ALL)` — необязательное поле с именами пользователей. Означает, что пользователь или пользователи могут выполнять команды от имени любого другого пользователя, указав его;
- `ALL` — в этом поле команды означает «любую команду», здесь можно перечислить только допустимые команды.

Дополнительная информация

- `man 8 sudo`
- `man 5 sudoers`

5.19. Увеличение времени кэширования пароля в sudo

Задача

В большинстве дистрибутивов Linux команда `sudo` кэширует пароли на 15 минут по умолчанию. Спустя это время вам придется вновь ввести свой пароль. Вас утомляет такой частый ввод пароля, когда приходится много работать, и хотелось бы увеличить продолжительность кэширования.

Решение

Измените продолжительность кэширования в файле `/etc/sudoers`. Откройте его для редактирования с помощью команды `visudo`:

```
$ sudo visudo
```

Затем найдите строку `Defaults` и задайте новую продолжительность кэширования. В следующем примере продолжительность кэширования увеличена до 60 минут:

```
$ Defaults timestamp_timeout=60
```

Если установить продолжительность равной 0, то `sudo` будет запрашивать пароль при каждом обращении к команде.

Если задать в `timestamp_timeout` отрицательное число, например, `-1`, то продолжительность кэширования пароля ограничиваться не будет.

Комментарий

Кэширование паролей в `sudo` является полезной защитой от ситуаций, когда вы можете забыть, что работаете с привилегиями root или отходите от компьютера и кто-то другой может захотеть развлечься с вашим компьютером.

Дополнительная информация

- `man 8 sudo`
- `man 5 sudoers`

5.20. Создание отдельных конфигураций для пользователей sudo

Задача

Определить разные конфигурации `sudo` для разных пользователей; например, задать для младших администраторов другое время кэширования пароля, отличающееся от вашего. У вас может быть настроена большая продолжительность кэширования, а вы хотите, чтобы у младших администраторов продолжительность была небольшой.

Решение

Отдельные конфигурации можно создавать в `/etc/sudoers.d`. Ниже представлен пример определения 30-минутного тайм-аута кэширования пароля для пользователя Stash:

```
$ cd /etc/sudoers.d/
$ sudo visudo -f stash
```

Введите **`Defaults timestamp_timeout=30`** и сохраните файл. После этого вы должны увидеть новый файл:

```
$ sudo ls /etc/sudoers.d/
README stash
```

В файлы отдельных конфигураций нужно вводить только настройки, отличающиеся от настроек в `/etc/sudoers`.

Комментарий

С помощью этой возможности удобно управлять несколькими пользователями. Вместо одного большого конфигурационного файла можно создать короткие файлы для каждого пользователя.

Дополнительная информация

- `man 8 sudo`
- `man 5 sudoers`

5.21. Управление паролем пользователя root

Задача

Вы были назначены системным администратором Linux с неограниченными привилегиями `sudo`, но вам забыли сообщить пароль root или сообщили, но вы позабыли его. Вам требуется задать новый пароль root.

Решение

Чтобы стать «настоящим» пользователем root, вызовите команду `su` с помощью `sudo`:

```
duchess@pc:~$ sudo su -l
[sudo] password for duchess:
root@pc:~#
```

После этого вы сможете выполнить команду `passwd` и задать новый пароль root.

Комментарий

Иногда команды `sudo` оказывается недостаточно и нужно ввести пароль root, например, после загрузки в аварийном режиме.

Дополнительная информация

- `man 8 sudo`
- `man 5 sudoers`
- `man 1 passwd`

5.22. Настройка sudo для использования без ввода пароля root

Задача

Организовать аутентификацию пользователей `sudo` их собственными паролями, если ваша система Linux запрашивает пароль пользователя root, как в следующем примере:

```
$ sudo visudo
[sudo] password for root:
```

Решение

Это поведение настроено по умолчанию в некоторых дистрибутивах Linux, таких как openSUSE.

Когда при установке Ubuntu Linux вы назначаете себя администратором, Ubuntu настраивает вашу учетную запись, наделяя ее неограниченными полномочиями `sudo`, эквивалентными полномочиям root, но с использованием вашего собственного пароля. В openSUSE, напротив, команда `sudo` настроена так, что требует ввода пароля целевого пользователя, каковым по умолчанию является root.

Чтобы настроить команду `sudo` для запроса собственных паролей пользователей, отредактируйте файл `/etc/sudoers`, закомментировав следующие две строки:

```
duchess@pc:~$ sudo visudo
# Defaults targetpw
# ALL ALL=(ALL) ALL
```

В openSUSE и Fedora создайте пользователей `sudo` с неограниченными привилегиями root, добавив их в группу `wheel` в файле `/etc/group`. (Чтобы узнать, как ограничить привилегии пользователей, см. рецепт 5.18.)

Изменение вступит в силу сразу после сохранения изменений и закрытия файла.

Комментарий

Защита пароля root — главная причина использования `sudo` вместо `su`.

Дополнительная информация

- Рецепт 5.18.

ГЛАВА 6

Управление файлами и каталогами

Linux предоставляет надежные базовые средства для управления доступом к файлам и каталогам в форме настраиваемых привилегий. Для всех файлов и каталогов поддерживается три уровня владения: пользователь, группа и все остальные; и несколько разрешений для доступа, включая чтение, запись и выполнение. Вы можете защитить свои личные файлы и определять, кто будет иметь к ним доступ, а пользователь root может управлять доступом к командам, сценариям, общим и системным файлам.

Даже при использовании более эффективных инструментов управления доступом, таких как SELinux или AppArmor, все равно важно знать базовые средства и понимать принципы их работы.

В системе Linux все пользователи — и люди, и системные службы — имеют учетные записи. Некоторым системным службам, как и людям, учетные записи необходимы для управления привилегиями.

Каждый файл имеет три уровня владения: владелец, группа и все остальные (иногда уровень *«все остальные»* называют *«мир»*). Владелец — это конкретный пользователь, группа-владелец — одна группа, а все остальные — это все, кто может получить доступ к файлу.

Каждый файл имеет шесть разрешений доступа — чтение, запись и выполнение — и три специальных режима: *sticky bit* (бит закрепления, или «липкий» бит), *setuid* и *setgid*.

Разрешения на доступ к файлу определяют, какие пользователи могут создавать, читать, редактировать или удалять файл, а также какие пользователи могут выполнять команды. Специальные разрешения определяют, кто может перемещать, удалять или переименовывать файл, а также кто может выполнять команду с повышенными привилегиями.

Разрешения на доступ к каталогу определяют, какие пользователи могут изменять имя каталога или входить в него, а также кто может читать, редактировать, добавлять или удалять файлы из каталога.

Запомните фундаментальный принцип безопасности Linux: назначайте минимальные разрешения, необходимые для выполнения работы.



Ограничения привилегий

Любой, имеющий разрешение на чтение файла, сможет его скопировать.

У вас не получится запретить доступ к вашим файлам пользователю root или пользователям sudo, обладающим достаточными привилегиями.

Разрешения и владение — это функции файловых систем, и их можно обойти, прочитав содержимое устройства хранения из другого экземпляра Linux, например загрузив Live-версию Linux со съемного носителя или переставив жесткий диск в другой компьютер. В этом случае вам понадобятся только права root в системе, к которой вы подключаете устройство хранения, и не нужно ничего знать об исходных владельцах файлов и разрешениях.

Пользователь root, которого также называют суперпользователем, безраздельно властвует в системе Linux. Он может делать практически все: редактировать и удалять файлы других пользователей, входить в любые каталоги и выполнять любые команды. Обычные (непривилегированные) пользователи могут временно получить полномочия root с помощью команд `sudo` или `su` (см. рецепты 5.17 и 5.18).

Каждый пользователь имеет уникальный UID и принадлежит как минимум к одной группе (см. рецепт 5.1). Каждый пользователь в группе автоматически получает разрешения этой группы.

Чтобы увидеть, как все это выглядит, загляните в каталог `/etc`, содержащий конфигурационные файлы системы:

```
$ stat --format=%a:%A:%U:%G /etc
755:drwxr-xr-x:root:root
```

Эта команда выводит *режим* каталога — набор разрешений — в двух формах: в форме восьмеричного числа (`755`) и в символической форме (`drwxr-xr-x`). Это два разных способа выражения одного и того же режима, который в этом примере определяет неограниченные привилегии для владельца каталога и только возможность входа в каталог для группы и всех остальных. Режимы файлов мы также подробно обсудим в данной главе.

Элемент `root:root` — владелец и группа. Владелец и группа могут различаться; например, каталог `/etc/cups` принадлежит пользователю `root` и группе `lp`.

В этой главе вы узнаете о специальных режимах: sticky bit, setuid и setgid. Режимы setuid и setgid повышают привилегии пользователей и групп до уровня пользователя и группы, владеющих файлом, соответственно. Они предназначены для особых случаев и должны применяться очень осторожно, поскольку повышение привилегий представляет потенциальную угрозу безопасности. Режим sticky bit не позволяет никому, кроме владельца или обладателя привилегий root, удалять, переименовывать или перемещать файлы в каталоге, которым они не владеют, например, в `/tmp`.

Далее вы узнаете, как управлять владением и режимами, создавать и удалять файлы и каталоги, настраивать привилегии по умолчанию, передавать права владения файлами другому пользователю или группе, а также копировать, перемещать и переименовывать файлы и каталоги.



Использование sudo

В большинстве примеров в этой главе используется приглашение к вводу в форме знака доллара, $, которое указывает на непривилегированного пользователя. В зависимости от уровня ваших привилегий вам может потребоваться использовать sudo для выполнения некоторых операций.

6.1. Создание файлов и каталогов

Задача

Организовать файлы, поместив их в каталоги.

Решение

Использовать команду `mkdir` для создания каталогов. Следующая команда создаст новый подкаталог в текущем каталоге:

```
$ mkdir -v presentations
mkdir: created directory 'presentations'
```

Создать двухуровневое дерево каталогов в текущем каталоге и использовать параметр `-p` (parent — «родитель») для создания вмещающего их каталога:

```
$ mkdir -p presentations/2020/august
mkdir: created directory 'presentations/2020'
mkdir: created directory 'presentations/2020/august'
```

Создать новый каталог верхнего уровня, указав путь относительно корня файловой системы, `/`. Для этого необходимы привилегии root:

```
$ sudo mkdir -v /charts
mkdir: created directory '/charts'
```

Одновременно с созданием каталога можно задать разрешения:

```
$ mkdir -m 0700 /home/duchess/dog-memes
```

Файлы создаются приложениями, например текстовыми процессорами, графическими редакторами и специальными командами, такими как `touch`. Она создает новый пустой файл:

```
$ touch newfile.txt
```

См. рецепт 6.2, в котором показано, как с помощью команды `touch` быстро создать пакет файлов для тестирования.

Комментарий

Если вам потребовалось получить визуальное представление дерева каталогов, попробуйте команду `tree`. Корень (`/`) дерева — вверху:

```
$ tree -L 1 /
/
├── backups
├── bin
├── boot
[...]
```

Вы, наверное, заметили, что это дерево перевернуто вверх ногами. В реальном мире деревья растут снизу вверх — от корня к ветвям, — но команда `tree` отображает дерево каталогов сверху вниз. На то есть причина: мы обычно читаем текст сверху вниз.

В этом примере перечислены только каталоги верхнего уровня в корневом каталоге. С параметром `-L 2` команда выведет также каталоги второго уровня, с параметром `-L 3` — три уровня и т. д.

Дополнительная информация

- Рецепт 6.2.
- `man 1 mkdir`
- `man 1 touch`
- `man 1 yes`
- `man 1 tree`

6.2. Быстрое создание пакетов файлов для тестирования

Задача

Создать пакет файлов для тестирования разрешений и вообще для любых других целей, когда может понадобиться множество файлов.

Решение

Использовать команду `touch`. Следующая команда создаст один пустой файл:

```
$ touch newfile.txt
```

А так можно создать 100 новых пустых файлов:

```
$ touch file{00..99}
```

Эта команда создаст 100 новых файлов с именами `file00`, `file01`, `file02` и т. д. При желании файлам можно дать расширение и выбрать любое другое имя:

```
$ touch test{00..99}.doc
$ ls
test00.doc
test01.doc
test02.doc
[...]
```

Если числа добавлять в начало имени, это упростит их упорядочение:

```
$ touch {00..99}test.doc
$ ls
00test.doc
01test.doc
02test.doc
[...]
```

Быстрый способ создать непустой файл — использовать команду `yes`. В следующем примере создается файл размером 500 Мбайт, заполненный повторяющейся строкой `This is a test file`:

```
$ yes This is a test file | head -c 500 MB > testfile.txt
```

Следующий пример демонстрирует создание 100 файлов, каждый размером 1 Мбайт:

```
$ for x in {01..100};
> do yes This is a test file | head -c 1MB > $x-testfile.txt;
> done
```

Список новых файлов выглядит следующим образом:

```
001-testfile.txt
002-testfile.txt
003-testfile.txt
[...]
```

Комментарий

Команды в этом рецепте позволяют настраивать имена и размеры файлов, нумерацию и текст для команды `yes`.

Примеры в рецепте дополняют числа в именах файлов ведущими нулями, чтобы они расположились в выводе команды `ls` в правильном порядке. Большинство графических диспетчеров файлов правильно обрабатывают нумерованные имена файлов, но по умолчанию команда `ls` использует лексикографический порядок. Следующий пример демонстрирует эту особенность с диапазоном нумерации от одной до трех цифр:

```
$ touch {0..150}test.doc
$ ls -C1
0test.doc
100test.doc
101test.doc
102test.doc
103test.doc
104test.doc
105test.doc
106test.doc
107test.doc
108test.doc
109test.doc
10test.doc
110test.doc
111test.doc
112test.doc
113test.doc
114test.doc
115test.doc
116test.doc
117test.doc
118test.doc
119test.doc
```

```
11test.doc
120test.doc
121test.doc
[...]
```

При лексикографическом упорядочении имена файлов интерпретируются как текстовые строки, а не как целые числа и символы, и каждая цифра и каждая буква сравнивается по отдельности слева направо. Алгоритм лексикографической сортировки не знает, что 10 меньше 100, и уверен, что 101 следует за 100, 102 следует за 101, а 10*t* следует за 109, поскольку буквы следуют за цифрами, то есть *t* следует за 9.

Чтобы избавиться от этого недостатка, можно добавить в числа ведущие нули, чтобы все они имели одинаковое количество цифр, или вывести список файлов с помощью команды `ls -v`. Она обрабатывает числа в именах файлов как целые числа, а не как символы и, соответственно, выводит имена файлов в правильном порядке.

Дополнительная информация

- `man 1 ls`
- `man 1 touch`
- `man 1 yes`

6.3. Относительные и абсолютные пути к файлам

Задача

Понять разницу между относительными и абсолютными путями к файлам и уметь определять свое текущее местоположение в файловой системе.

Решение

Абсолютные пути к файлам всегда начинаются с корня, `/`, например `/boot` или `/etc`. Относительные пути к файлам откладываются от текущего каталога и не имеют косой черты в начале. Предположим, вы находитесь в своем домашнем каталоге и он содержит следующие подкаталоги:

```
madmax@client2:~$ ls --group-directories-first
Audiobooks
```

```
bin
Desktop
Documents
Downloads
games
Music
Pictures
Public
Templates
Videos
```

В этом примере абсолютный путь к каталогу `Audiobooks` будет иметь вид `/home/madmax/Audiobooks`, а относительный путь — `Audiobooks`. Ниже показано, как с помощью команды `cd` можно перейти в этот каталог, используя абсолютный путь:

```
$ cd /home/madmax/Audiobooks
```

и относительный путь:

```
$ cd Audiobooks
```

Каталог, в котором вы находитесь, называется текущим рабочим. Узнать свой текущий рабочий каталог можно с помощью команды `pwd` (print working directory — «напечатать рабочий каталог»):

```
$ pwd
/home/madmax
```

Комментарий

Абсолютные и относительные пути к файлам — частый источник путаницы. Запомните: если путь к файлу начинается с косой черты (`/`), то является абсолютным. Если не с косой — относительным, который откладывается от текущего рабочего каталога.

Некоторые приложения и команды требуют указывать относительные пути; например, в списках включения и исключения в команде `rsync` используются пути к файлам, откладываемые относительно копируемых каталогов.

Дополнительная информация

- `man 1 pwd`
- Глава 7.

6.4. Удаление файлов и каталогов

Задача

Забавы ради было создано множество файлов и каталогов, а теперь их нужно удалить.

Решение

Использовать команду `rm` (remove — «удалить»), соблюдая меры предосторожности, — она с радостью удалит все, что вы ей передадите, поэтому передавайте ей только файлы и каталоги, которые действительно нужно удалить.

Удаление единственного файла с подробным отчетом о выполняемых действиях выглядит так:

```
$ rm -v aria.ogg
removed 'aria.ogg'
```

Флаг `-i` потребует подтвердить решение перед удалением:

```
$ rm -iv intermezzo.wav
rm: remove regular file 'intermezzo.wav'? y
removed 'intermezzo.wav'
```

Добавьте флаг `-r` (recursive — «рекурсивно»), чтобы удалить каталог и все файлы и каталоги, содержащиеся в нем. Комбинация параметров `-r` и `-i` заставит вас подтвердить каждое удаление:

```
$ rm -rvi rehearsals
rm: descend into directory 'rehearsals'? y
rm: remove regular file 'rehearsals/brass-section'? y
[...]
```

Если вы абсолютно уверены в своем решении и не желаете подтверждать каждое удаление, то опустите параметр `-i`.

Следующая команда удалит только подкаталог `jan`:

```
$ rm -rv rehearsals/2020/jan
```

Следующая команда удалит каталог `rehearsals` и все файлы и каталоги в нем:

```
$ rm -rv rehearsals
```

В именах файлов можно использовать подстановочные (шаблонные) символы, например, удалить все файлы с определенным расширением можно следующим образом:

```
$ rm -v *.txt
```

или с именами, начинающимися с определенной последовательности символов:

```
$ rm -v aria*
```

Если `rm` отказывается удалить файл или каталог и вы уверены, что хотите его удалить, то добавьте параметр `-f` (force — «принудительно»).

Комментарий

Команда `rm -rf /` сотрет всю корневую файловую систему (конечно, если вы обладаете привилегиями root). Некоторые считают смешным посоветовать новичкам выполнить данную команду. На самом деле здесь нет ничего смешного. Забавно запустить ее на тестовой или на виртуальной машине и понаблюдать, как долго система продолжит работать после этого, поскольку процессы в памяти продолжат выполняться даже после того, как файловая система будет стерта с диска.

Дополнительная информация

- `man 1 rm`

6.5. Копирование, перемещение и переименование файлов и каталогов

Задача

У вас есть файлы и каталоги, и нужно переместить файлы в каталоги, изменить имена каких-то файлов и сделать копии.

Решение

Использовать команду `cp` для копирования и команду `mv` для перемещения или переименования.

Ниже представлен пример копирования двух файлов из текущего рабочего каталога в каталог `~/songs2`:

```
$ cp -v aria.ogg solo.flac ~/songs2/
'aria.ogg' -> '/home/duchess/songs2/aria.ogg'
'solo.flac' -> '/home/duchess/songs2/solo.flac'
```



Тильда представляет ваш домашний каталог

Знак тильды (~) служит кратким представлением пути к вашему домашнему каталогу, то есть в примере выше ~/songs2 означает то же, что и /home/duchess/songs2/.

Копирование каталога со всем содержимым можно выполнить, добавив параметр `-r` (recursive — «рекурсивно»):

```
$ cp -rv ~/music/songs2 /shared/archives
```

Этот рекурсивный пример просто скопирует каталог `songs2` с файлами. Если добавить параметр `--parents`, то команда сохранит структуру родительских каталогов. Следующий пример скопирует каталог `songs1` с его содержимым и сохранит путь `duchess/music/songs2/`:

```
$ cp -rv --parents duchess/music/songs2/ shows/
duchess -> shows/duchess
duchess/music -> shows/duchess/music
'duchess/music/songs2' -> 'shows/duchess/music/songs2'
'duchess/music/songs2/intro.flac' -> 'shows/duchess/music/songs2/intro.flac'
'duchess/music/songs2/reprise.flac' -> 'shows/duchess/music/songs2/reprise.flac'
'duchess/music/songs2/solo.flac' -> 'shows/duchess/music/songs2/solo.flac'
```

Другое содержимое каталогов `duchess` и `music` скопировано не будет — только каталог `songs2` с его содержимым.

Команда `mv` перемещает и переименовывает файлы. Ниже представлен пример перемещения двух файлов в другой каталог:

```
$ mv -v aria.ogg solo.flac ~/songs2/
renamed 'aria.ogg' -> '/home/duchess/songs2/aria.ogg'
renamed 'solo.flac' -> '/home/duchess/songs2/solo.flac'
```

А эта команда переместит каталог в другой каталог:

```
$ mv -v ~/songs2/ ~/music/
```

Комментарий

Ниже представлены некоторые полезные параметры команды `cp`:

- `-a`, `--archive` — сохраняет все атрибуты файлов, такие как разрешения, владение и отметки времени;
- `-i`, `--interactive` — запрашивает подтверждение, если копирование приведет к затиранию существующих файлов;
- `-u`, `--update` — затирает существующие файлы, только если копируемый файл более новый. Этот параметр поможет сэкономить время, когда требуется повторно скопировать пакет файлов, часть из которых не изменилась (`rsync` справляется с этой задачей эффективнее, копируя только изменения, см. главу 7).

Команда `mv` тоже имеет несколько полезных параметров:

- `-i`, `--interactive` — запрашивает подтверждение, если перемещение приведет к затиранию существующих файлов;
- `-n`, `--no-clobber` — предотвращает затирание существующих файлов;
- `-u`, `--update` — перемещает файлы, только если они более новые, чем файлы в каталоге назначения, или если они перемещаются впервые.

Дополнительная информация

- `man 1 cp`
- `man 1 mv`

6.6. Настройка разрешений файлов с помощью команды chmod с использованием восьмеричного представления

Задача

Вы уже знаете, что команда `chmod` (change mode — «изменить режим») поддерживает как восьмеричное, так и символьное представление разрешений, и хотели бы использовать восьмеричное представление для управления правами доступа к файлам.

Решение

В следующих примерах показано, как назначить различные разрешения для файлов, используя восьмеричное представление. Первый пример дает доступ для чтения и записи владельцу файла `file.txt` и исключает любой доступ для группы и всех остальных (мира):

```
$ chmod -v 0600 file.txt
mode of 'file.txt' changed from 0644 (rw-r--r--) to 0600
(rw-------)
```

Владелец файла сможет читать и редактировать файл, а также удалить его. Другие пользователи ничего из этого не смогут, даже прочитать файл, хотя смогут видеть его в диспетчере файлов.

Если разрешить чтение и запись миру, то любой сможет сделать с файлом все что угодно:

```
$ chmod 0666 file.txt
```

Следующая команда дает доступ для чтения/записи к файлу `file.txt` его владельцу и только для чтения группе и всем остальным:

```
$ chmod -v 0644 file.txt
mode of 'file.txt' changed from 0666 (rw-rw-rw-) to 0644 (rw-r--r--)
```

Часто используется набор разрешений, предоставляющий владельцу и группе одинаковые права, такие как права на чтение и запись, и запрещающий доступ всем остальным (миру):

```
$ chmod 0660 file.txt
```

Команды и сценарии должны иметь разрешение на выполнение. Следующая команда делает сценарий `backup.sh` выполняемым и доступным для чтения и записи владельцу, выполняемым и доступным для чтения группе и недоступным для всех остальных:

```
$ chmod 0750 backup.sh
```

В восьмеричном представлении доступно четыре поля, но вы чаще будете использовать последние три и очень редко — первое поле. Первое поле зарезервировано для специальных режимов (см. рецепт 6.8).

Комментарий

В восьмеричном представлении используются целые числа 0–7. В табл. 6.1 показаны отношения между владельцами и разрешениями.

Таблица 6.1. Восьмеричные значения полей

Режим	Владелец	Группа	Остальные
Чтение	4	4	4
Запись	2	2	2
Выполнение	1	1	1
Отсутствие разрешений	0	0	0

У файла или каталога есть один владелец-пользователь и владелец-группа. *Остальные* — это все остальные. Каталог или выполняемый файл, доступ к которым не ограничен для всех, имеет режим 0777, а неограниченный файл — режим 0666.

Если вы не знакомы с правами доступа к файлам в Linux, то вам может быть полезно увидеть их в другом представлении, например, в табл. 6.2.

Таблица 6.2. Разрешения для файлов в Linux

Разрешение	Описание
7	Чтение, запись, выполнение. Каталоги отличаются от файлов, поскольку для всех каталогов требуется установить бит выполнения. Вы можете назначить каталогу любые разрешения, как и файлу, но без установленного бита выполнения никто не сможет войти в каталог (командой cd или в диспетчере файлов). Сценарии и двоичные команды тоже должны иметь установленный бит выполнения, иначе они будут интерпретироваться как обычные файлы
6	Чтение и запись
5	Чтение и выполнение. Это типичный набор разрешений для команд
4	Чтение
3	Запись и выполнение
2	Запись
1	Выполнение
0	Полное отсутствие разрешений

Дополнительная информация

- `man 1 chmod`
- Рецепт 6.8.

6.7. Настройка разрешений каталогов с помощью команды chmod с использованием восьмеричного представления

Задача

Вы уже знаете, что разрешения для каталогов настраиваются иначе, чем для файлов, и хотели бы использовать восьмеричное представление для управления ими.

Решение

В наборе разрешений для каталогов всегда должен быть установлен бит выполнения. Это может показаться странным, но бит, управляющий разрешением на выполнение, позволяет входить[1] в каталог с помощью команды `cd` или в диспетчере файлов.

Следующая команда создаст каталог `shared`:

```
$ sudo mkdir /shared
```

А эта команда откроет доступ для чтения/записи владельцу каталога `/shared` и только для чтения всем остальным:

```
$ chmod 0755 /shared
```

Владелец имеет неограниченные права доступа к каталогу. Группа и все остальные (мир) смогут входить в каталог и читать файлы, но не смогут их редактировать или добавлять.

Следующая команда применит одни и те же разрешения к существующему содержимому каталога благодаря параметру `-R` (recursive — «рекурсивно»):

```
$ chmod -R 0755 /shared
```

Следующая команда ограничит доступ к каталогу и его существующему содержимому, разрешив его только владельцу каталога. Файлы и под-

---

[1] Точнее говоря, получить список содержимого каталога.

каталоги, находящиеся в нем, могут принадлежать другим владельцам и иметь другие разрешения, но останутся недоступными для группы и всех остальных (мира):

```
$ chmod 0700 /shared
```

На практике обычно используется набор разрешений, предоставляющий владельцу и группе одинаковые права, такие как «чтение-запись-выполнение», и запрещающий доступ всем остальным:

```
$ chmod 0770 /shared
```

Комментарий

Группы и каталоги открывают широкие возможности управления доступом к файлам. Создавайте группы в соответствии с функциями; например, у разных групп могут быть свои общие каталоги. В большинстве случаев не требуется сверхтонкое разграничение, и во многих организациях и их подразделениях предпочитают не использовать жесткие ограничения. Какими бы ни были ваши потребности, старая команда chmod по-прежнему остается основным инструментом для управления разрешениями на доступ к файлам.

Дополнительная информация

- man 1 chmod

6.8. Особые режимы для особых случаев использования

Задача

Установить некоторые разрешения, не поддерживаемые традиционным набором разрешений «владелец-группа-остальные», например разрешить непривилегированным пользователям запускать команду, требующую повышенных привилегий, защитить файлы в каталоге, который является совместным для нескольких пользователей, или принудительно применить определенные разрешения к файлам в каталоге.

Решение

Задействуйте особые режимы sticky bit, setuid и setgid (табл. 6.3). Режим sticky bit применяется к каталогам с файлами, принадлежащими разным пользователям, чтобы не позволить пользователям перемещать, переименовывать или удалять файлы, которыми они не владеют:

```
$ chmod -v 1770 /home/duchess/shared
mode of '/home/duchess/shared changed from 0770 (rwxrwx---) to 1770 (rwxrwx--T)
```

Режим setuid применяется к выполняемым файлам, чтобы повысить привилегии пользователя, запускающего этот файл, до уровня привилегий владельца файла:

```
$ chmod 4750 backup-script
mode of 'backup-script' changed from 0750 (rwxrw----) to 4770 (rwsrwx---)
```

Режим setgid применяется к каталогам: все вновь создаваемые файлы в этом каталоге будут принадлежать к той же группе, что и сам он. Это отличный прием, позволяющий определить верные уровни владения файлами в общем каталоге:

```
$ chmod 2770 /home/duchess/shared
mode of '/home/duchess/shared' changed from 0770 (rwxrwx---) to 2770 (rwxrws---)
```

Режим setgid также может применяться к выполняемым файлам, чтобы изменить эффективную группу на группу, владеющую файлом.

Комментарий

Режимы setgid и setuid способны создавать бреши в системе безопасности, которыми могут воспользоваться злоумышленники или неблагонадежные пользователи. Их следует использовать, только когда не получается придумать более безопасный способ добиться желаемого, например, с применением разрешений для групп или `sudo`.

Режим setuid применяется к выполняемым файлам.

Режим setgid применяется к каталогам и выполняемым файлам.

Режим sticky bit применяется только к каталогам. В табл. 6.3 показаны отношения между владельцами и разрешениями.

Специальные режимы можно комбинировать (табл. 6.4).

Режим sticky bit (бит закрепления, или «липкий» бит) имеет описательное название: *бит ограничения возможности удаления*. Этот бит запрещает непри-

вилегированным пользователям удалять или переименовывать файл в каталоге, если он не принадлежит им. Вы можете увидеть действие данного режима в своем каталоге `/tmp`, который доступен для чтения и записи и содержит файлы, принадлежащие нескольким пользователям. Применение режима sticky bit к этому каталогу не позволяет пользователям перемещать, переименовывать или удалять файлы, которыми они не владеют, даже если у них есть права на запись для некоторых таких файлов:

```
$ stat --format=%a:%A:%U:%G /tmp
1777:drwxrwxrwt:root:root
```

Таблица 6.3. Восьмеричные значения полей

Режим	Специальные режимы	Владелец	Группа	Остальные
Чтение		4	4	4
Запись		2	2	2
Выполнение		1	1	1
setuid	4			
setgid	2			
sticky bit	1			
Отсутствие разрешений	0	0	0	0

Таблица 6.4. Значения sticky bit, setgid, setuid

Режим	Восьмеричное значение
Все биты сброшены	0
Установлен sticky bit	1
Установлен setgid	2
Установлены sticky bit и setgid	3
Установлен setuid	4
Установлены sticky bit и setuid	5
Установлены setgid и setuid	6
Установлены sticky bit, setgid и setuid	7

Режиму sticky bit здесь соответствует цифра 1 в 1777.

Название setgid расшифровывается как set group user identification («установить идентификатор группы пользователей»), а setuid — как set user identification

(«установить идентификатор пользователя»). Они используются для повышения привилегий пользователей до уровня пользователя или группы, владеющего (-ей) файлом. Благодаря биту setuid непривилегированные пользователи могут использовать команду `passwd` для изменения своих паролей, даже притом, что только root имеет право на запись в `/etc/passwd`, а все остальные могут только читать его:

```
$ stat --format=%a:%A:%U:%G /usr/bin/passwd
4755:-rwsr-xr-x:root:root
```

Цифра 4 в наборе разрешений 4755 для файла `/usr/bin/passwd` — это бит setuid, который означает, что команда `passwd`, запущенная любым пользователем, будет выполняться с привилегиями root.

Дополнительная информация

- `man 1 chmod`

6.9. Удаление особых режимов с помощью восьмеричного представления

Задача

Удалить особые режимы из набора разрешений для файла или каталога.

Решение

Удаление особых режимов несколько отличается от их установки, поскольку при этом требуется использовать дополнительный ведущий ноль, как в следующем примере:

```
$ chmod -v 00770 backup.sh
mode of 'backup.sh' changed from 1770 (rwxrwx--T) to 0770 (rwxrwx---)
```

Ведущий ноль можно заменить знаком равенства:

```
$ chmod -v =770 backup.sh
mode of 'backup.sh' changed from 1770 (rwxrwx--T) to 0770 (rwxrwx---)
```

Дополнительная информация

- `man 1 chmod`

6.10. Настройка разрешений файлов с помощью команды chmod с использованием символического представления

Задача

Вы уже знаете, что команда `chmod` (change mode — «изменить режим») поддерживает как восьмеричное, так и символьное представление разрешений, и хотели бы использовать символическое представление для управления правами доступа к файлам.

Решение

Символическое представление сложнее восьмеричного, и его поведение зависит от используемого оператора.

Поддерживаются три оператора: `+`, `-` и `=`. Вы можете изменить разрешения сразу для всех уровней владения, использовав флаг `a`, или по отдельности, использовав флаг `u` для обозначения владельца, `g` — группы и `o` — всех остальных:

- `+` добавляет указанное разрешение в набор;
- `-` удаляет указанное разрешение из набора;
- `=` добавляет указанные разрешения и удаляет неуказанные.

Допустим, что файл `file.txt` доступен владельцу для чтения и записи, группе для чтения и всем остальным для чтения, то есть имеет режим `-rw-r--r--`:

```
$ stat --format=%a:%A:%U:%G file.txt
664:-rw-r--r--:stash:stash
```

Требуется изменить набор разрешений, чтобы получить `-rw-rw-rw-`, то есть добавить разрешение на запись для группы и всех остальных. Это можно сделать следующим образом:

```
$ chmod -v g+w,o+w file.txt
mode of 'file.txt' changed from 0644 (rw-r--r--) to 0666 (rw-rw-rw-)
```

Кроме того, можно использовать параметр `a=rw`.

В следующем примере владелец файла изменяет режим доступа к файлу `file.txt` так, что он становится доступным для чтения и записи только владельцу, а члены группы и все остальные смогут только читать его:

```
$ chmod -v g-w,o-w file.txt
mode of 'file.txt' changed from 0666 (rw-rw-rw-) to 0644 (rw-r--r--)
```

Часто используется набор разрешений, предоставляющий владельцу и группе одинаковые права, такие как права на чтение и запись, и запрещающий доступ всем остальным (миру):

```
$ chmod -v u=rw,g=rw,o-r file.txt
mode of 'file.txt' changed from 0644 (rw-r-r--) to 0660 (rw-rw----)
```

Команды и сценарии должны иметь разрешение на выполнение. Следующая команда добавляет бит выполнения для владельца в существующий набор разрешений:

```
$ chmod -v u+x file.txt
mode of 'file.sh' changed from 0660 (rw-rw----) to 0760 (rwxrw----)
```

Оператор = удобно использовать для затирания существующих разрешений:

```
$ chmod -v u=rw,g=rw,o=r file.txt
mode of 'file.sh' changed from 0760 (rwxrw----) to 0664 (rw-rw-r--)
```

Комментарий

Символическое представление разрешений в команде `chmod` обеспечивает явность намерений и простоту сохранения существующих разрешений. Используйте операторы добавления и удаления разрешений из существующего набора (кроме оператора =, который затирает существующий набор) и укажите флаг `u`, `g`, `o` или `a`.

Символическое представление специально разрабатывалось для простоты запоминания, `r` — read («чтение»), `w` — write («запись») и `x` — execute («выполнение») (табл. 6.5).

Таблица 6.5. Символические представления разрешений

Режим	Значение
к	read — чтение
w	write — запись
x	execute — выполнение

Символические обозначения уровней владения также легко запоминаются (табл. 6.6).

Символическое представление, подобно восьмеричному, поддерживает также специальные режимы (см. рецепт 6.11).

Таблица 6.6. Символические представления уровней владения

Уровень владения	Обозначение
Пользователь (user)	r
Группа (group)	g
Все остальные (other)	o
Все (all)	a

Ниже представлено десятизначное символическое представление набора разрешений для домашнего каталога пользователя Duchess (минусы обозначают неустановленные, то есть отсутствующие, разрешения):

```
$ stat --format=%a:%A:%U:%G /home/duchess
755:drwxr-xr-x:duchess:duchess
```

Символ `d` в наборе `drwxr-xr-x` сообщает, что это каталог (directory). В восьмеричном представлении нет аналогичного обозначения.

Остальные девять символов делятся на три триады, и три символа в каждой триаде представляют разрешения для чтения, записи и выполнения.

Дополнительная информация

- `man 1 chmod`

6.11. Настройка особых режимов с помощью команды chmod с использованием символического представления

Задача

Установить особые режимы с помощью команды `chmod` с использованием символического представления.

Решение

К особым режимам относятся sticky bit, setuid и setgid. Все они задаются в полях разрешения для выполнения. (См. конец подраздела «Комментарий» в рецепте 6.10, если вы забыли, что это за поля.)

Примените режим sticky bit к каталогам с файлами, принадлежащими разным пользователям, чтобы не позволить пользователям перемещать, переименовывать или удалять файлы, которыми они не владеют:

```
$ chmod o+t /shared/stickydir
mode of '/shared/stickydir' changed from 0775 (rwxrwxr-x) to 1775 (rwxrwxr-t)
```

Примените режим setgid к каталогу, чтобы все вновь создаваемые файлы в нем принадлежали той же группе, что и сам каталог. Это отличный прием, позволяющий определить верные уровни владения файлами в общем каталоге:

```
$ chmod -v g+s /shared
mode of '/shared' changed from 0770 (rwxrwx---) to 2770 (rwxrws---)
```

Примените режим setuid к выполняемым файлам, чтобы позволить непривилегированным пользователям выполнять привилегированные операции:

```
$ chmod -v u+s backup-script
mode of 'backup-script' changed from 0755 (rwxr-xr-x) to 4755 (rwsr-xr-x)
```

Режимы setgid и setuid могут создавать бреши в системе безопасности; см. подраздел «Комментарий», в котором приводится дополнительная информация.

Комментарий

Режим setuid применяется к выполняемым файлам.

Режим setgid применяется к каталогам и выполняемым файлам.

Режим sticky bit применяется только к каталогам.

В табл. 6.7 показаны отношения между владельцами и разрешениями.

Таблица 6.7. Все символические обозначения режимов

Режим	Владелец	Группа	Остальные
Чтение	r	r	r
Запись	w	w	w
Выполнение	x	x	x
setuid	s		
setgid		s	
sticky bit			t

Режим sticky bit (бит закрепления, или «липкий» бит) имеет описательное название: *бит ограничения возможности удаления*. Он запрещает непривилегированным пользователям удалять или переименовывать файл в каталоге, если он

не принадлежит им. Вы можете увидеть действие этого режима в своем каталоге `/tmp`, который доступен для чтения и записи и содержит файлы, принадлежащие нескольким пользователям. Применение режима sticky bit к данному каталогу не позволяет пользователям перемещать, переименовывать или удалять файлы, которыми они не владеют:

```
$ stat --format=%a:%A:%U:%G /tmp
1777:drwxrwxrwt/:root:root
```

Название setgid расшифровывается как set group user identification («установить идентификатор группы пользователей»), a setuid — как set user identification («установить идентификатор пользователя»). Они служат для повышения привилегий пользователей до уровня пользователя или группы, владеющего (-ей) файлом. Благодаря биту setuid непривилегированные пользователи могут с помощью команды `passwd` менять свои пароли, даже притом, что только root имеет право делать запись в файл `/etc/passwd`, а все остальные могут только читать его:

```
$ stat --format=%a:%A:%U:%G /usr/bin/passwd
4755:-rwsr-xr-x:root:root
```

Элемент `rws` в поле разрешений для владельца означает доступность файла для чтения и записи владельцу, а также для выполнения всем пользователям с привилегиями владельца.

Режимы setgid и setuid могут создавать бреши в системе безопасности. Их следует использовать, только когда не получается придумать более безопасный способ добиться желаемого, например с помощью разрешений для групп или `sudo`.

Дополнительная информация

- `man 1 chmod`

6.12. Настройка разрешений для групп файлов с помощью команды chmod

Задача

Установить определенный набор разрешений сразу для нескольких файлов.

Решение

Команда `chmod` поддерживает операции со списками файлов. Кроме того, можно использовать команду `find` и подстановочные (шаблонные) символы командной оболочки для выбора файлов, режимы доступа к которым нужно изменить.



Вам может понадобиться команда sudo

Если вы увидите сообщение Permission denied (Доступ запрещен), то используйте sudo.

Следующая команда принимает список файлов, перечисленных через пробел, и во всех устанавливает режим доступа «только для чтения» для всех:

```
$ chmod -v 444 file1 file2 file3
```

Чтобы установить разрешения для каталога и его содержимого, добавьте параметр `-R` (recursive — «рекурсивно»):

```
$ chmod -vR 755 /shared
```

Для выбора файлов можно использовать подстановочные (шаблонные) символы; следующая команда выберет все файлы `.txt` в текущем каталоге и сделает их доступными для чтения и записи владельцу и только для чтения — группе и всем остальным:

```
$ chmod -v 644 *.txt
```

Подстановочный (шаблонный) символ можно использовать для выбора всех файлов, имена которых начинаются с одной и той же последовательности:

```
$ chmod -v 644 abcd*
```

Следующая команда сделает все файлы в текущем каталоге доступными для чтения и записи владельцу и группе, оставив разрешения для каталога без изменений:

```
$ find . -type f -exec chmod -v 660 {} \;
```

Можно изменить разрешения для всех файлов, принадлежащих определенному пользователю, указав его имя или числовой идентификатор. Именно это делает следующий пример, начинающий поиск от корня файловой системы:

```
$ sudo find / -user madmax -exec chmod -v 660 {} \;
$ sudo find / -user 1007 -exec chmod -v 660 {} \;
```

Комментарий

Для поиска во всех каталогах вам понадобятся привилегии root.

Точка (`find .`) сообщает команде `find`, что она должна начать поиск с текущего каталога. При необходимости поиск можно начать с любого каталога.

`-type f` требует выполнять поиск только среди файлов.

`-user` проверяет принадлежность файлов указанному пользователю.

`-exec chmod -v 660 {} \;` — это чудесное маленькое заклинание, которое берет результаты поиска и применяет к ним команду `chmod -v 660`. В этом месте можно использовать практически любую команду, какую вы решите применить к результатам поиска.

Дополнительная информация

- `man 1 chmod`
- `man 1 find`

6.13. Настройка владения файлами и каталогами с помощью команды chown

Задача

Сменить владельца файла или каталога.

Решение

Используйте команду `chown` (change owner — «сменить владельца»), чтобы сменить владельца файла. В простейшем виде эта команда имеет следующий синтаксис: `chown` *`пользователь:группа имя_файла`*. Можно сменить и только владельца: `chown` *`пользователь: имя_файла`* — или только группу: `chown` *`:группа имя_файла`*.

Для смены владельца нужны привилегии root:

```
duchess@client1:~$ sudo chown -v madmax: song.wav
changed ownership of 'song.wav' from duchess:duchess to madmax:duchess
```

Смена группы:

```
$ sudo chown -v :composers song.wav
changed ownership of 'song.wav' from madmax:duchess to :composers
```

Смена и владельца, и группы:

```
$ sudo chown stash:stash song.wav
```

Комментарий

Чтобы передать файл, не принадлежащий вам, во владение другому пользователю, нужны привилегии root. Смена группы может обойтись без привилегий root, если файл принадлежит вам и вы входите в состав обеих групп.

При смене только владельца двоеточие можно опустить, но при смене группы двоеточие обязательно должно присутствовать.

Дополнительная информация

- `man 1 chown`

6.14. Смена владельца для групп файлов с помощью команды chown

Задача

Сменить владельца для каталога и всего его содержимого, или только для содержимого каталога — списка файлов, или для файлов, принадлежащих определенному пользователю.

Решение

Команда `chown` поддерживает операции со списками файлов. Кроме того, можно использовать команду `find` и подстановочные (шаблонные) символы командной оболочки для выбора файлов, владение которыми нужно изменить.

Чтобы сменить владельца у нескольких файлов одной командой `chown`, перечислите эти файлы через пробел:

```
$ sudo chown -v madmax:share file1 file2 file3
```

Следующая команда передаст все файлы с расширением `.txt`, находящиеся в текущем каталоге, во владение новой группе:

```
$ sudo chown -v :share *.txt
```

Следующая команда сменит владельца у всех файлов в указанном каталоге, принадлежащих указанному пользователю; пользователей можно указывать по именам или числовым идентификаторам:

```
$ chown -Rv --from duchess stash /shared/compositions

$ chown -Rv --from 1001 1005 /shared/compositions
```

Для смены владельца заданных файлов во всей файловой системе или в определенном каталоге и его подкаталогах используйте команду `find`:

```
$ sudo find / -user duchess -exec chown -v stash {} \;

$ sudo find / -user 1001 -exec chown -v 1005 {} \;
```

Комментарий

Передача владения всеми файлами от одного пользователя другому или от одной группы другой может пригодиться в случае удаления учетной записи пользователя.

Дополнительная информация

- `man 1 chown`

6.15. Настройка разрешений по умолчанию с помощью команды umask

Задача

Файлы создаются с некоторым стандартным набором разрешений по умолчанию. Как задать эти разрешения?

Решение

Разрешения по умолчанию определяются маской `umask` (user file-creation mode mask — «маска режимов по умолчанию для файлов, вновь создаваемых пользователем»). Узнать текущее состояние маски можно с помощью одноименной команды:

```
$ umask
0002
```

Та же маска в символическом представлении выглядит следующим образом:

```
$ umask -S
u=rwx,g=rwx,o=rx
```

Она устанавливает набор разрешений по умолчанию 0775 для каталогов и 0664 для файлов, то есть `umask` «маскирует» жестко заданные разрешения по умолчанию 0777 и 0666. Операцию маскировки можно также представить как операцию вычитания (удаления разрешения), 0777 – 0002 = 0775.

Чтобы изменить маску на время, до конца текущего сеанса, введите команду `umask` с новым значением маски, например:

```
$ umask 0022
```

Чтобы новая маска использовалась постоянно, вставьте команду `umask 0022` в свой файл `~/.bashrc`.

Маска по умолчанию для всех пользователей в системе назначается в файле `/etc/login.defs`:

```
UMASK 022
```

В табл. 6.8 вы увидите некоторые типичные значения `umask`.

Комментарий

Команда `umask` является встроенной командой Bash, а не выполняемой программой где-нибудь в `/bin`, `/usr/bin` или где-то еще.

Некоторые часто используемые значения `umask` перечислены в табл. 6.8.

Таблица 6.8. Типичные значения umask

umask	Каталоги	Файлы
0002	0775	0664
0022	0755	0644
0007	0770	0660
0077	0700	0600

Дополнительная информация

- `man 1 chmod`
- Дополнительную информацию о команде `umask` и других встроенных командах Bash можно найти в разделе Shell Builtin Commands (Встроенные команды оболочки) руководства `man 1 bash`.

6.16. Создание символических и жестких ссылок на файлы и каталоги

Задача

Создать ссылки на файлы.

Решение

В Linux есть два типа ссылок: символические и жесткие. Символические ссылки могут ссылаться на файлы и каталоги, а жесткие — только на файлы.

Ссылки обоих типов создаются командой `ln` (link — «ссылка»). Следующая команда создаст в текущем каталоге символическую ссылку на внешний каталог `/files/userstuff`:

```
$ ln -s /files/userstuff stuff
```

Здесь `/files/userstuff` — это цель, а `stuff` — имя символической ссылки. Вы можете давать своим ссылкам любые имена, перемещать и удалять их, не оказывая влияния на их цели. При попытке открыть символическую ссылку он будет вести себя подобно своей цели.

Жесткие ссылки — это копии файлов. По умолчанию команда `ln` создает жесткие ссылки:

```
$ ln /files/config1.txt myconf.txt
```

Комментарий

Символические ссылки могут ссылаться на файлы и каталоги, а жесткие ссылки — только на файлы.

Символические ссылки

Символические ссылки ссылаются на файлы и каталоги. Когда цель символической ссылки удаляется, переименовывается или перемещается, символическая ссылка перестает работать. В случае создания нового файла с тем же именем, что и у удаленного файла, символическая ссылка восстановит работоспособность, даже если содержимое файла будет другим.

Символические ссылки могут пересекать границы файловых систем. Вы можете даже создавать символические ссылки на файлы или каталоги, недоступные постоянно, например находящиеся на USB-накопителе или на сетевом файловом ресурсе.

Символические ссылки не обновляются при изменении цели (переименовании, перемещении или удалении). Вам нужно создавать новую символическую ссылку и удалять старую.

Разрешения или владельца символической ссылки нельзя изменить, поскольку они зависят от разрешений целевого объекта.

Символическая ссылка выглядит следующим образом:

```
$ stat stuff
  File: stuff -> /files/userstuff
  Size: 4               Blocks: 0        IO Block: 4096   symbolic link
Device: 804h/2052d      Inode: 877581    Links: 1
Access: (0777/lrwxrwxrwx) Uid: ( 1000/ madmax) Gid: ( 1000/ madmax)
```

Строка `File: stuff → /files/userstuff` показывает цель, на которую ссылается символическая ссылка.

Третья строка сообщает, что это символическая ссылка (symbolic link).

Символ `l` в строке `Access: lrwxrwxrwx` также сообщает, что это символическая ссылка.

Ниже показано, как выглядит символическая ссылка в выводе команды `ls`:

```
$ ls -l
[...]
lrwxrwxrwx 1 madmax madmax 4 Apr 26 12:42 stuff -> /files/userstuff
```

Жесткие ссылки

Каждый файл уникально идентифицируется *индексным узлом* (inode), а индексный узел — это то, на что ссылается жесткая ссылка, то есть жесткая ссылка ссылается не на имя файла, а на индексный узел. Индексные узлы можно увидеть в выводе команды `ls`, если запустить ее с параметром `-i`. В следующем примере индексный узел имеет номер 1353, и тот же номер используется в трех жестких ссылках:

```
$ ls -li
1353 -rw-rw-r-- 3 madmax madmax 11208 Apr 26 13:06 config.txt
1353 -rw-rw-r-- 3 madmax madmax 11208 Apr 26 13:06 config2.txt
1353 -rw-rw-r-- 3 madmax madmax 11208 Apr 26 13:06 config3.txt
```

То есть все три индексных узла указывают на один и тот же блок данных.

Жесткие ссылки остаются работоспособными всегда, поскольку ссылаются непосредственно на индексные узлы. Файлы, на которые ссылается несколько жестких ссылок, можно перемещать, переименовывать и редактировать, и все эти операции будут отражаться на жестких ссылках, поскольку все они ссылаются на один и тот же блок данных.

Для каждого файла в Linux создается жесткая ссылка. Создавая жесткую ссылку, вы фактически создаете новое имя файла для существующего блока данных.

Жесткие ссылки не могут пересекать границ файловых систем и существуют только в рамках одной файловой системы. Например, если `/` и `/home` находятся в разных дисковых разделах, то вы не сможете создать жесткую ссылку в `/home`, ссылающуюся на файл в `/`.

Можно создать сколько угодно жестких ссылок, указывающих на один и тот же файл, и дисковое пространство, занятое данными, на которые указывают эти ссылки, не изменится.

Сравните жесткие ссылки с копиями файлов: каждая копия занимает столько же места на диске, сколько занимает оригинал, каждая копия независима и копии можно отправить куда угодно.

Файл не удаляется с диска полностью, пока не будут удалены все жесткие ссылки на него. Убедиться в этом можно с помощью команды `ls`. Ниже показано продолжение предыдущего примера с тремя жесткими ссылками:

```
$ stat config3.txt
  File: config3.txt
  Size: 11208          Blocks: 24        IO Block: 4096   regular file
Device: 804h/2052d     Inode: 1353       Links: 3
```

Сравните значения полей `File`, `Size` и `Links` со значениями этих же полей в примерах с символическими ссылками. Жесткая ссылка — обычный файл, обратите внимание на поле `Links: 3`. Оно показывает, что в данном случае есть три жесткие ссылки, указывающие на одни и те же данные. При удалении файла с несколькими жесткими ссылками он не удаляется, пока не будет удалена последняя жесткая ссылка. Ниже показано, как можно найти все связанные жесткие ссылки с помощью команды `find`:

```
$ find /etc -xdev -samefile config3.txt
./config
./config2
./config3
```

Символические ссылки широко применяются в Linux, жесткие ссылки — намного реже. Некоторые приложения резервного копирования задействуют индексные узлы для обнаружения дубликатов. В старину, когда файловые системы были намного меньше, исчерпание индексных узлов не было редкостью, и в ту пору жесткие ссылки были предпочтительнее, поскольку каждая символическая ссылка имеет собственный индексный узел, а жесткие ссылки используют один общий индексный узел.

Узнать количество индексных узлов в файловой системе и сколько из них занято, можно с помощью команды `du`:

```
$ df -i /dev/sda4
Filesystem        Inodes  IUsed     IFree IUse% Mounted on
/dev/sda4      384061120 389965 383671155    1% /home
```

С 1 % занятых индексных узлов я нескоро исчерпаю их.

Дополнительная информация

- `man 1 ls`

6.17. Сокрытие файлов и каталогов

Задача

Скрыть некоторые каталоги и файлы, чтобы никто не увидел их.

Решение

Чтобы скрыть файлы от постороннего взгляда, поместите их на устройство хранения, доступное только вам.

Чтобы уменьшить беспорядок в диспетчере файлов, используйте *файлы с именами, начинающимися с точки*. У вас уже есть такие файлы. Поищите настройку Show hidden files (Показывать скрытые файлы) в своем диспетчере файлов или используйте команду `ls` с параметром `-a`:

```
$ ls -a
.
..
Audiobooks
.bash_history
.bash_logout
.bashrc
```

```
bin
.bogofilter
.cache
Calibre-Library
cat-memes
.cddb
.cert
```

Добавление точки в начало имени любого файла делает его скрытым, хотя на самом деле он не скрывается, а просто игнорируется, пока вы сами не захотите его увидеть. Этот прием используется в основном в домашних каталогах пользователей, чтобы уменьшить беспорядок за счет сокрытия конфигурационных файлов. Это самые обычные файлы, которые можно редактировать, удалять и выполнять с ними другие операции.

Комментарий

Обратите внимание на одинарные и двойные точки в начале списка файлов. Одна точка представляет текущий каталог, а две — родительский. Попробуйте передать их команде `cd`. После первой команды вы останетесь в текущем каталоге, а после второй перейдете в родительский:

```
stash@client4:~$ cd .
stash@client4:~$

stash@client4:~$ cd ..
stash@client4:/home$
```

Выполните команду `cd` без параметров, чтобы вернуться в свой домашний каталог, или `cd -`, чтобы вернуться в предыдущий каталог.

Дополнительная информация

Дополнительную информацию о команде `cd` и других встроенных командах Bash можно найти в разделе Shell Builtin Commands (Встроенные команды оболочки) руководства `man 1 bash`.

ГЛАВА 7

Резервное копирование и восстановление с помощью команд rsync и cp

Всем известно, насколько важно делать своевременные резервные копии файлов и периодически проверять их, чтобы увидеть, можно ли восстановить файлы из резервной копии. Но как это сделать в Linux? Не волнуйтесь, резервное копирование и восстановление файлов в Linux выполняются легко и просто, а сами резервные копии несложно найти и восстановить.

Полезно иметь пару USB-накопителей, чтобы попрактиковаться с командами, описанными в этой главе, и создать несколько каталогов, заполненных файлами, которые, как предполагается, не должны быть потеряны ни в коем случае.

Мы будем использовать команды `rsync` и `cp`. Обе являются важными инструментами в Linux, и вы можете рассчитывать, что они будут доступны всегда.

Команда `cp` входит в пакет GNU coreutils, который по умолчанию устанавливается почти во всех дистрибутивах Linux. Она предназначена для простого копирования. Ее может оказаться вполне достаточно для регулярного создания резервных копий.

Команда `rsync` — эффективный инструмент для передачи файлов. Ее основная цель — синхронизировать две файловые системы. При использовании этой команды для резервного копирования можно обеспечить синхронизацию локальных файлов с файлами на вашем устройстве для резервного копирования. Она работает быстро и эффективно, поскольку передает только изменения в файлах. В отличие от многих программ резервного копирования, которые препятствуют удалению чего бы то ни было, `rsync` способна отразить удаления файлов в резервной копии. Благодаря своим возможностям эта команда считается предпочтительным инструментом для обновления и зеркалирования до-

машних каталогов пользователей, сайтов, репозиториев GIT и других больших и сложных деревьев каталогов с файлами.

Есть два способа использовать команду `rsync` в сети: через SSH, для аутентифицированного входа в систему и передачи файлов, или путем запуска в режиме демона. При работе через SSH необходимо, чтобы пользователи имели учетные записи на каждой машине, к которой они собираются обращаться через `rsync`. В случае, когда команда запускается в режиме демона, можно использовать встроенные методы аутентификации для управления доступом, поэтому пользователям не нужны учетные записи на сервере rsync. Режим демона хорошо подходит для организации сервера резервного копирования в локальной сети. Доступ через ненадежные сети небезопасен, если не использовать VPN (см. главу 13).

На каком устройстве предпочтительнее хранить резервные копии? Это зависит от ваших потребностей. Я предпочитаю USB-накопители, когда речь идет о резервном копировании файлов одного пользователя. Предположим, у вас есть настольный компьютер с Linux, ноутбук, планшет и смартфон. Создайте резервную копию телефона и планшета на ПК, а затем сделайте резервную копию ПК на жестком диске, подключаемом к USB. Сверхважные файлы можно отправить в сетевую службу резервного копирования.

Для резервного копирования файлов нескольких пользователей хорошим решением может быть центральный сервер резервного копирования — любой компьютер с Linux на борту.

Учитывайте продолжительность хранения резервных копий. С цифровыми носителями нельзя рассчитывать на долговечность, поскольку, даже если носитель (жесткий диск, USB-накопитель, CD/DVD) доживет до часа X, нет никаких гарантий, что инструменты для его чтения сохранятся. Меняются оборудование и форматы файлов. У вас сохранилась возможность читать дискеты? А помните ZIP-диски? А как насчет тех старых архивов с документами Microsoft Word и PowerPoint? Имея файлы в форматах с открытым исходным кодом, всегда можно найти способ их восстановить. С проприетарными форматами все намного хуже, поскольку производитель может решить прекратить их поддержку.

Бумага до сих пор остается чемпионом по долговечности хранения, и этот факт следует учитывать, принимая решение о сохранении важных документов и фотографий.

Чтобы увеличить долговечность резервных копий «в цифре», запланируйте периодический перенос архивов на новые носители, возможно, с применением новых команд и в новых форматах. А как насчет резервного копирования самого резервного сервера? Да без проблем! Настройка удаленного зеркала rsync для резервного копирования резервных копий — распространенная стратегия, если интернет-соединение достаточно надежно для передачи массивного трафика.

Но прежде чем создавать массивную инфраструктуру резервного копирования, подумайте, сколько уровней избыточности вам нужно на самом деле. Внешние хранилища резервных копий — страховка от катастрофы в вашем вычислительном центре. Роль такого внешнего хранилища может играть удаленный сервер резервного копирования в вашем вычислительном центре, который контролируется вами, или в вычислительном центре друга. Это также может быть арендованный сервер в коммерческом центре обработки данных. Может быть, вам хватит и регулярного сброса данных на внешний жесткий диск, который затем помещается в банковский сейф. Но, помимо резервного копирования, подумайте также о восстановлении: насколько быстро можно получить доступ к резервным копиям?

Всегда помните: цель резервного копирования — *восстановление*. Регулярно проверяйте свои резервные копии, чтобы не узнать на собственном опыте, что ваш метод резервного копирования никуда не годится.

В текущей главе вы узнаете о простом копировании на USB-накопители с помощью `cp`. Некоторым пользователям этого более чем достаточно.

Большая часть главы посвящена применению команды `rsync` для быстрого и эффективного копирования. Команду `rsync` можно использовать для резервного копирования файлов на локальные носители или на удаленные серверы. Здесь вы узнаете, какие файлы желательно копировать, как настроить выбор файлов, сохранить права доступа к файлам и их временные метки, создать сервер резервного копирования rsync для нескольких пользователей и как обезопасить резервное копирование на удаленный сервер.

7.1. Выбор файлов для резервного копирования

Задача

Определить, для каких файлов следует создавать резервные копии. Нужно ли копировать какие-то системные файлы? Нужно ли копировать все свои личные файлы? Есть ли файлы, для которых не имеет смысла создавать резервные копии?

Решение

Любой файл, который жалко потерять, — это файл, заслуживающий внимания в смысле резервного копирования. Это могут быть и ваши личные файлы, и системные. Резервное копирование системных файлов, таких как команды, при-

ложения и библиотеки, едва ли стоит производить, поскольку их всегда можно загрузить и переустановить.

Резервного копирования заслуживают каталоги, содержащие конфигурационные файлы; файлы с данными веб-, FTP- и почтовых серверов; файлы журналов; приложения, установленные в нестандартные каталоги; и общие каталоги, такие как:

- `/boot/grub`, если в нем содержатся нестандартные файлы, такие как темы, фоновые изображения или шрифты;
- `/etc`, содержащий системные конфигурационные файлы;
- `/home`, включающий личные файлы пользователей;
- `/mnt`, содержащий временные точки монтирования файловых систем. Копируйте этот каталог, если у вас есть точки монтирования, которые хотелось бы сохранить;
- `/opt`, хранящий лицензионное или другое программное обеспечение, которое не устанавливается стандартным способом;
- `/root`, содержащий личные файлы пользователя root;
- `/srv`, включающий данные веб-, ftp- и rsync-серверов;
- `/tmp`, содержащий временные данные, которые автоматически обновляются или удаляются по мере необходимости. Отдельные данные в `/tmp` хранятся постоянно, например файлы, созданные пользователями или системными службами, и для них тоже следует создавать резервные копии;
- `/var`, хранящий самые разные типы данных, такие как файлы журналов, почтовые архивы, задания cron и данные системных служб, хотя большинство дистрибутивов перешли на использование `/srv` в системных службах.

Если у вас есть какие-то общие каталоги, нестандартные команды и сценарии, любые файлы с данными или каталоги, не перечисленные выше, то копируйте их.

Псевдофайловые системы `/proc`, `/sys` и `/dev` существуют только в оперативной памяти, и их не следует копировать.

`/media` предназначен для монтирования съемных носителей и управляется системой, поэтому его тоже не следует копировать. Если вы вручную создаете точки монтирования в `/media`, то для этой цели лучше использовать каталог `/mnt`.

Базы данных не следует копировать с помощью простых инструментов, поскольку для создания резервных копий баз данных и восстановления из них есть специальные утилиты и процедуры. Используйте эти инструменты. Примерами таких баз данных могут служить PostrgreSQL, MariaDB и MySQL.



Восстановление из резервных копий

Некоторые файлы не должны восстанавливаться из резервных копий (см. рецепт 7.2).

Самый простой путь — копировать все подряд, если ваше хранилище резервных копий достаточно велико для этого. Кроме того, можно пойти по пути тонкой настройки и составить список файлов для копирования или исключений (см. рецепты 7.8 и 7.9).

Комментарий

Носители данных сейчас настолько дешевы, что вам, возможно, не придется заботиться об экономии места. Но если вы вынуждены учитывать ограничения хранилища, см. рецепты в этой главе, описывающие выбор файлов.

Дополнительная информация

- Стандартная иерархия файловой системы (https://oreil.ly/y1pJs).

7.2. Выбор файлов для восстановления из резервной копии

Задача

Вы решили восстановить файлы из резервной копии и хотите узнать, какие файлы не следует восстанавливать.

Решение

Некоторые файлы, в зависимости от обстоятельств, не следует восстанавливать. Не восстанавливайте файл `/etc/fstab` после переустановки Linux (он содержит настройки монтирования статических файловых систем). Каждый раз, когда устанавливается Linux, все файловые системы получают новые универсальные уникальные идентификаторы (Universal Unique Identifier, UUID), поэтому они не будут опознаны и установка завершится ошибкой.

Будьте осторожны, восстанавливая любые файлы из `/etc` или файлы с именами, начинающимися с точки (например, `/home/.config` или `/home/.local`), в своем домашнем каталоге. Если вы восстанавливаете файлы из резервной копии сразу после установки другого дистрибутива или другой версии того же дистрибутива

Linux, то могут возникнуть проблемы с несовместимостью в параметрах конфигурации или расположения файлов. Восстанавливайте их по одному, чтобы можно было быстро обнаружить любые проблемы.

Дополнительная информация

- Глава 1.

7.3. Простейший метод создания локальной резервной копии

Задача

Выяснить самый простой и легкий способ регулярного создания резервных копий на локальном запоминающем устройстве USB.

Решение

Обратитесь к простому копированию. Купите себе хороший жесткий диск USB (или флешку). Подключите его и используйте диспетчер файлов для копирования. Легко, быстро, без суеты, да и восстановить файлы из такой резервной копии очень просто. Или прибегните к команде `cp` (см. рецепт 7.4).

Комментарий

Простое копирование плохо масштабируется, но для некоторых устройств, таких как персональный компьютер, ноутбук и телефон, его более чем достаточно. Самое главное — не забывать регулярно делать резервные копии, проверять возможность восстановления файлов из таких резервных копий и не беспокоиться о том, не слишком ли вы занудны.

Раньше резервное копирование было более сложной задачей, поскольку устройства хранения стоили дорого и программы резервного копирования использовали множество хитроумных уловок для экономии места. Теперь вполне можно купить внешний жесткий диск USB 3.0 емкостью в пару терабайт менее чем за 200 долларов.

Дополнительная информация

- `man 1 cp`

7.4. Автоматизация создания локальной резервной копии

Задача

Вам понравилось с помощью простого приема копирования создавать резервные копии на внешнем USB-накопителе и теперь хотелось бы автоматизировать этот процесс.

Решение

Используйте команду `cp` и `crontab` для планирования резервного копирования.

Файлы и каталоги для резервного копирования можно перечислить прямо в команде `cp`, разделяя их пробелами:

```
duchess@pc:~$ cp -auv Pictures/cat-desk.jpg Pictures/cat-chair.png \
  ~/cat-pics /media/duchess/2tbdisk/backups/
```

Следующая команда скопирует весь домашний каталог пользователя Duchess в каталог `backups` на внешнем USB-накопителе с именем `2tbdisk`:

```
duchess@pc:~$ cp -auv ~ /media/duchess/2tbdisk/backups/
```

Она создаст каталог `/media/duchess/2tbdisk/backups/duchess/` на устройстве резервного копирования.

А так можно скопировать содержимое каталога, не копируя сам каталог:

```
duchess@pc:~$ cp -auv /home/duchess/* /media/duchess/2tbdisk/backups/
```

Ниже представлен пример создания задания в cron, которое будет создавать резервные копии по вечерам в 22:30:

```
duchess@pc:~$ crontab -e
# m h  dom mon dow   command
30 22  *   *   *   /bin/cp -au /home/duchess /media/duchess/2tbdisk/backups/
```

Комментарий

Чтобы сохранить атрибуты файла, такие как право собственности и разрешения, отформатируйте диск для резервных копий в файловой системе, которая поддерживает такие атрибуты, например Ext4, XFS или Btrfs (см. главу 11). Файловые системы FAT не поддерживают права собственности или разрешения.

Определите, сколько времени потребуется для резервного копирования. Если это займет больше времени, чем запланированный интервал, то cron запустит следующее резервное копирование по расписанию, и тогда могут возникнуть проблемы.

Первый запуск занимает больше всего времени, поскольку все файлы новые. Последующие сеансы резервного копирования будут выполняться быстрее, так как будут копироваться только новые файлы и файлы с более новыми отметками времени.

Тильда (~) — это сокращенное представление пути к домашнему каталогу текущего пользователя, то есть в данном рецепте тильда представляет каталог `/home/duchess`.

Звездочка в `/home/duchess/*` означает «все файлы в `/home/duchess`», но не включает сам каталог `/home/duchess`.

Параметры `-a`, `-u` и `-v` команды `cp` означают:

- `-a`, `--archive` — рекурсивно копировать файлы с сохранением всех атрибутов: режима, владения, отметок времени и других дополнительных атрибутов;
- `-u`, `--update` — копировать только изменившиеся и новые файлы;
- `-v`, `--verbose` — выводить сообщения в процессе копирования, помогающие видеть, как протекает процесс копирования.

Ниже представлены еще несколько полезных параметров:

- `-R`, `-r` — рекурсивно; используйте этот параметр для копирования каталогов, когда не применяется параметр `-a`, который гарантирует сохранность атрибутов файла, а `-R`, `-r` — нет. Файловые системы FAT и exFAT не поддерживают атрибуты файлов, поэтому используйте `-R`, `-r` при работе с ними;
- `--parents` — создает отсутствующие родительские каталоги на устройстве для резервного копирования;
- `-x`, `--one-file-system` — предотвращает рекурсивный обход других дисковых разделов и смонтированных сетевых файловых систем. Например, если у вас смонтирован общий том NFS, то, возможно, вы не захотите добавлять его в резервную копию.

Большинство дистрибутивов Linux монтируют USB-устройства в `/run/media` или `/media`. Самый простой способ найти путь к USB-накопителю — заглянуть в диспетчер файлов или использовать команду `lsblk`:

```
$ lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
[...]
sdb      8:16   0  1.8T  0 disk
└─sdb1   8:17   0  1.5T  0 part /media/duchess/backups
```

Дополнительная информация

- Рецепт 3.7, в котором описывается планировщик заданий cron.
- `man 1 crontab`
- `man 1 cp`

7.5. Использование команды rsync для создания локальной резервной копии

Задача

Организовать создание резервных копий на USB-накопителе или жестком диске USB, но желательно использовать что-то более быстрое и эффективное, чем простая команда `cp`. Вдобавок нужен простой способ восстановления файлов с помощью стандартных инструментов Linux, то есть без специального программного обеспечения.

Решение

Используйте команду `rsync`. Она обеспечивает синхронизацию файловых систем, как локальных, так и удаленных. Это быстрый и эффективный инструмент, поскольку передает только изменения в файлах и позволяет восстанавливать файлы с помощью самой себя, команды `cp`, диспетчера файлов или любого другого инструмента копирования, который вы предпочитаете.

Следующий пример показывает, как создать резервную копию домашнего каталога. Сначала укажите исходный каталог, то есть тот, который требуется скопировать, а затем каталог для резервной копии. В этом примере копируется домашний каталог пользователя Duchess на USB-накопитель с именем `2tbdisk`:

```
duchess@pc:~$ rsync -av ~ /media/duchess/2tbdisk/
sending incremental file list
duchess/
duchess/Documents/
duchess/Downloads/
duchess/Music/
[...]
```

```
sent 27,708,209 bytes received 20,948 bytes 11,091,662.80 bytes/sec
total size is 785,103,770,793 speedup is 28,313.29
```

Можно указать два и более каталога для передачи в резервную копию, перечислив их через пробел:

```
duchess@pc:~$ rsync -av ~/arias ~/overtures /media/duchess/2tbdisk/duchess/
```

Чтобы восстановить файлы из резервной копии, достаточно поменять местами исходный и конечный каталоги:

```
duchess@pc:~$ rsync -av /media/duchess/2tbdisk/duchess/arias /home/duchess/
```

Безопасно протестировать команду `rsync` без копирования файлов можно, добавив параметр `--dry-run`:

```
duchess@pc:~$ rsync -av --dry-run \
~/Music/scores ~/Music/woodwinds /media/duchess/2tbdisk/duchess/
```

Если какие-то файлы будут удалены из исходного каталога, то `rsync` не удалит их из резервной копии, но об этом можно явно попросить, добавив параметр `--delete`:

```
duchess@pc:~$ rsync -av --delete /home/duchess /media/duchess/2tbdisk/
```

Комментарий

Тильда (~) — это сокращенное представление пути к домашнему каталогу текущего пользователя, то есть в данном рецепте тильда представляет каталог `/home/duchess`.

В примерах команд с разрывами строки используется обратная косая черта (\), указывающая, что команда продолжается на следующей строке. Вы можете скопировать всю команду с обратной косой чертой, и она должна выполниться правильно.

Если на вашем компьютере смонтированы сетевые файловые системы, такие как NFS или Samba, то используйте параметр `-x`, чтобы копировать только файлы из локальной файловой системы, без рекурсивного обхода удаленных файловых систем.

При добавлении косой черты в конце `~/` (`/home/duchess/`) будет копироваться только содержимое каталога `duchess/`, но не сам он, в результате получится набор файлов `/media/duchess/2tbdisk/[`*файлы*`]`. Если опустить косую черту в конце, то файлы из `/home/duchess` будут скопированы вместе с каталогом

`duchess` и получится: `/media/duchess/2tbdisk/duchess/[`*файлы*`]`. Завершающая косая черта имеет значение только для исходного каталога и не имеет значения для целевого.

Не расстраивайтесь, если вам придется считать на пальцах или выполнять много тестовых прогонов, чтобы понять, как ведет себя завершающая косая черта, — все проходят через это. Косую черту в конце можно представлять себе как небольшой забор, который не позволяет исходному каталогу проскользнуть в резервную копию.

Параметры `-a` и `-v` команды `rsync` означают следующее:

- `-a`, `--archive` — сохранить режим, отметки времени, разрешения, атрибуты владения и выполнить копирование рекурсивно. Этот параметр действует подобно набору параметров `-rlptgoD`, которые заставляют `rsync` копировать рекурсивно, копировать символические ссылки, сохранять разрешения, сохранять время последнего изменения файла, сохранять атрибуты владения и копировать специальные файлы, такие как файлы устройств;
- `-v`, `--verbose` — выводить сообщения, помогающие видеть, как протекает процесс.

Ниже представлены еще несколько полезных параметров:

- `-q`, `--quiet` — подавить вывод любых сообщений, кроме сообщений об ошибках;
- `--progress` — показывать информацию о процессе передачи каждого файла;
- `-A`, `--als` — сохранять списки управления доступом (Access Control List, ACL);
- `-X`, `--xattrs` — сохранять дополнительные атрибуты файлов (xattrs).

Имена ваших файлов, конечно, будут отличаться от приведенных в примерах. `2tbdisk` — это метка файловой системы, созданная пользователем (см. рецепт 9.4), — сокращение от 2 terabyte disk (двухтерабайтный диск). Если вы не создадите метку сами, то `udev` создаст ее за вас, например: `/media/duchess/488B-7971/`.

Для восстановления файлов из резервной копии можно использовать обычные инструменты Linux, такие как `rsync`, диспетчер файлов или команда `cp`.

Дополнительная информация

- `man 1 rsync`

7.6. Безопасная передача файлов с помощью rsync по сети через SSH

Задача

Использовать `rsync` для копирования файлов на другой компьютер в локальной сети или через Интернет с помощью транспорта с шифрованием и аутентификацией.

Решение

При передаче файлов на другой компьютер `rsync` по умолчанию использует SSH. На удаленной машине должен быть запущен SSH-сервер, а на исходной машине — настроен SSH-клиент (см. главу 12).

Ниже представлен пример передачи файлов по локальной сети с персонального компьютера на ноутбук. На компьютере пользователь зарегистрирован как Duchess, а на ноутбуке — как Empress. Файлы из домашнего каталога на персональном компьютере копируются в домашний каталог на ноутбуке:

```
duchess@pc:~$ rsync -av ~/Music/arias empress@laptop:songs/
duchess@laptop's password:
building file list ... done
arias/
arias/o-mio-babbino-caro.ogg
arias/deh-vieni-non-tardar.ogg
arias/mi-chiamano-mimi.ogg
wrote 25984 bytes read 68 bytes 7443.43 bytes/sec
total size is 25666 speedup is 0.99
```

Если каталог назначения отсутствует, то `rsync` создаст его.

Для передачи файлов через Интернет используйте полное доменное имя сервера, куда вы собираетесь копировать файлы:

```
duchess@pc:~$ rsync -av ~/Music/woodwinds \
  empress@remote.example.com:/backups/
```

Чтобы скопировать файлы в обратную сторону, источник и приемник следует поменять местами. Ниже представлен пример копирования каталога `/woodwinds` со всем его содержимым с удаленного хоста в домашний каталог Duchess:

```
duchess@pc:~$ rsync -av empress@remote.example.com:/backups/woodwinds \
  /home/duchess/Music/
```

Комментарий

Было время, когда параметр `ssh` требовалось передавать явно, например: `rsync -a -e ssh [`*параметры*`]`. Теперь в этом нет необходимости.

Вам могут пригодиться некоторые из следующих параметров:

- `--partial` сохраняет частично скачанные файлы, когда сетевое соединение неожиданно разрывается, и возобновляет передачу с того места, где она была прервана, после восстановления соединения;
- `-h`, `--human-readable` отображает размеры файлов в килобайтах, мегабайтах и гигабайтах, а не в байтах;
- `--log-file=` записывает полный отчет о каждой передаче в указанном текстовом файле. Использовать все эти параметры вместе можно следующим образом:

```
duchess@pc:~$ rsync --partial --progress \
--log-file=/home/duchess/rsynclog.txt \
-hav ~/Music/arias empress@remote.example.com:/backups/
```

Аутентификация и шифрование берет на себя протокол SSH. Пользователям нужны учетные записи на всех машинах, на которые они собираются передавать файлы. См. главу 12, чтобы узнать о безопасном удаленном администрировании с помощью SSH.

Подумайте о возможности установки центрального сервера для хранения резервных копий, чтобы упростить администрирование. У ваших пользователей будут свои учетные записи со своими домашними каталогами на сервере, и они смогут управлять своими резервными копиями, не беспокоя вас.

Другой вариант организации сервера резервного копирования — запустить `rsync` как службу. Преимущество данного подхода в том, что вашим пользователям не понадобятся учетные записи на сервере. Но есть и недостаток: в таком случае они лишатся поддержки шифрования. См. рецепт 7.13, в котором об этом рассказывается подробнее.

Дополнительная информация

- `man 1 rsync`
- Рецепт 12.5.
- Рецепт 12.7.

7.7. Автоматизация резервного копирования с помощью rsync, cron и SSH

Задача

Создать задание cron для автоматического и безопасного резервного копирования с помощью `rsync`.

Решение

Для решения поставленной задачи необходимо настроить на целевом компьютере SSH с аутентификацией без пароля (см. рецепты 12.10 и 12.11) и сетевой доступ к целевому компьютеру для клиентов.

Затем можно использовать `/etc/crontab` для передачи, требующей привилегий root. Следующий пример копирует файлы из `/etc` каждый вечер в 22:00 на сервер с именем `server1`, находящийся в локальной сети:

```
# m h dom mon dow user  command
00 22 * * * root /usr/bin/rsync -a /etc server1:/system-backups
```

Для передачи личных файлов можно использовать персональные crontab (см. рецепт 3.7).

Комментарий

OpenSSH — прекрасный инструмент, обеспечивающий безопасную передачу данных по сети и пригодный для множества задач. Все, что обменивается информацией по сети, вероятно, сможет работать через SSH.

Дополнительная информация

- Глава 12.
- Рецепт 12.10.
- Рецепт 12.11.

7.8. Исключение файлов из резервного копирования

Задача

До сих пор во всех примерах демонстрировалось, как передавать каталоги целиком. Но иногда желательно исключить из резервного копирования некоторые файлы и каталоги.

Решение

Для простоты следующие примеры демонстрируют копирование на USB-накопитель, но их также можно приспособить для передачи на удаленный компьютер по SSH (см. рецепт 7.6).

Если файлов, которые нужно исключить, немного, то их можно перечислить в командной строке с помощью параметра `--exclude=`. Ниже представлен пример исключения одного файла из каталога `/home/duchess/Music/arias`:

```
duchess@pc:~$ rsync -av --exclude=lho-perduta.wav \
  ~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

Красиво, просто и надежно. Однако есть один подвох: если в исходном каталоге имеется несколько файлов с именем, указанным в параметре `--exclude=`, то все они будут исключены из копирования. Если нежелательно исключать дубликаты, то следует указать, какой конкретно нужно исключить. Следующий пример демонстрирует, как исключить из копирования только файл в исходном каталоге `arias/`:

```
duchess@pc:~$ rsync -av --exclude=arias/lho-perduta.wav \
~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

Чтобы исключить несколько файлов, заключите их в фигурные скобки, разделив одинарными кавычками и запятыми. Между знаком равенства и фигурной скобкой не должно быть пробелов, а также между запятыми и одинарными кавычками:

```
duchess@pc:~$ rsync -av \
--exclude={'arias/lho-perduta.wav','non-mi-dir.wav','un-bel-di-vedremo.flac'} \
~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

Каталоги исключаются точно так же, как и файлы, а кроме того, допускается смешивать файлы и каталоги в одном списке:

```
duchess@pc:~$ rsync -av \
--exclude={'soprano/','tenor/','non-mi-dir.wav'} \
~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

См. рецепт 7.11, где показано, как поместить список исключений в файл.

Комментарий

Корневой каталог в команде `rsync` — это каталог верхнего уровня, из которого копируются файлы. В примерах выше это `~/Music/arias`. `rsync` проверяет все файлы и каталоги в вашем корневом каталоге и сравнивает их с исключениями, которые в терминологии `rsync` называются *образцами*. Образцы сравниваются с именами файлов и каталогов в исходном каталоге, начиная от корня и далее вниз по иерархии каталогов. Каждый раз, когда образец совпадает с именем файла, этот файл исключается из передачи. Если обнаруживается совпадение с образцом `arias/lho-perduta.wav` где-то в другом месте, например `2arias/lhoperduta.wav`, то данный файл также будет исключен. Когда образец заканчивается косой чертой (`/`), `rsync` будет сопоставлять его только с каталогами.

Дополнительная информация

- `man 1 rsync`
- Рецепт 7.10.

7.9. Выборочное включение файлов в резервное копирование

Задача

Вместо определения списка исключений нужно определить список файлов для включения в резервную копию.

Решение

Если требуется скопировать лишь несколько файлов, то их можно перечислить прямо в командной строке. Параметр `--include=` действует иначе, чем `--exclude=`, в том смысле, что в действительности он означает не «включить» (include), а «не исключать». Фактически параметр `--include` должен использоваться в комплексе

с двумя дополнительными параметрами `--include=*/` и `--exclude='*'`, как демонстрирует следующий пример копирования единственного файла:

```
duchess@pc:~$ rsync -av --include=*/ --include=lho-perduta.wav \
 --exclude='*' ~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

В параметре можно передать список файлов:

```
duchess@pc:~$ rsync -av --include=*/ \
--include={'lho-perduta.wav','non-mi-dir.wav','un-bel-di-vedremo.flac'} \
--exclude='*' ~/Music/arias /media/duchess/2tbdisk/duchess/Music/
```

Между знаком равенства и фигурной скобкой не должно быть пробелов, а также между запятыми и одинарными кавычками.

Если в исходном каталоге окажется несколько файлов с именем, указанным в параметре `--include`, то `rsync` скопирует их все. Следующий пример скопирует только файл `/home/duchess/Music/arias/sopranos/lho-perduta.wav`, поскольку образец `soprano/lhoperduta.wav` является уникальным в `/Music/arias`:

```
duchess@pc:~$ rsync -av --include=*/ --include=soprano/lho-perduta.wav
--exclude='*' ~/Music/arias /media/duchess/2tbdisk/duchess/Music/
Music/
Music/arias/
Music/arias/baritone/
Music/arias/soprano/
Music/arias/soprano/lho-perduta.wav
Music/arias/tenor/
[...]
```

Этот пример скопирует только один файл, но все подкаталоги в `~/Music/arias`. Чтобы исключить копирование пустых каталогов, используйте параметр `-m`, `--prune-empty-dirs`, как в следующем примере:

```
duchess@pc:~$ rsync -avm --include=*/ --include=soprano/lho-perduta.wav
--exclude='*' ~/Music/arias /media/duchess/2tbdisk/duchess/Music/
Music/
Music/arias/soprano/
Music/arias/soprano/lho-perduta.wav
```

Если список файлов для включения слишком длинный, то запишите его в простой текстовый файл (см. рецепты 7.10 и 7.11).

Комментарий

Параметр `--include=*/` требует от `rsync` выполнить обход всего исходного каталога.

Выражение `--include=[файлы]` означает «не исключать эти файлы».

Параметр `--exclude='*'` требует от `rsync` исключить все, что не включено.

Запомните, что все пути, используемые в команде, откладываются относительно исходного каталога, а не от корня файловой системы.

Дополнительная информация

- `man 1 rsync`
- Рецепт 7.10.
- Рецепт 7.11.

7.10. Управление включением с помощью простого файла со списком для включения

Задача

Список файлов для включения слишком длинный, чтобы набирать его в командной строке, и хотелось бы сохранить его в файле и каким-то образом передать его команде `rsync`. Кроме того, учитывая прошлый опыт применения `--include`/`--exclude`, хотелось бы, чтобы команда с использованием такого файла была максимально простой.

Решение

Самый простой способ передать список файлов, не сходя с ума от синтаксиса `--include`/`--exclude` в `rsync`, — создать простой файл со списком и передать его в параметре `--files-from=`. Вам не стоит беспокоиться о том, чтобы расположить файлы в правильном порядке или использовать нотацию фильтра в `rsync`, требуется лишь самый простой список с любыми файлами и каталогами, которые нужно скопировать. Единственная проблема — каждый элемент в этом списке должен выражать полный путь от исходного каталога. В следующем примере все элементы списка — это пути от каталога `/home/duchess`:

```
# файл со списком для включения
#
/Documents/compositions/jazz/
/Documents/schedule.odt
```

```
/Videos/concerts/
.config
.local
/Music/courses/bassoon.avi</strong>
[...]
```

Этот файл можно передать в параметре `--files-from`:

```
duchess@pc:~$ rsync -av ~ --files-from ~/include-list.txt \
 duchess@remote.example.com:/backups/
```

Комментарий

Это самый простой способ управления списком файлов и каталогов для резервного копирования. Нет никаких исключений, никаких подстановочных знаков, никакого замысловатого синтаксиса, просто ясный и понятный список.

При использовании тильды (~) для обозначения домашнего каталога не ставьте знак равенства в параметре `--files-from`, как показано в последнем примере.

Дополнительная информация

- `man 1 rsync`
- Рецепт 7.9.
- Рецепт 7.11.

7.11. Управление включением и исключением с помощью файла со списком для исключения

Задача

Вам понравилась простая идея файла со списком для включения, изложенная в рецепте 7.10, но теперь хотелось бы иметь возможность управлять списком не только включаемых, но и исключаемых файлов.

Решение

Используйте файл со списком для исключения. Такой файл предлагает больше гибкости и может содержать как включаемые, так и исключаемые файлы. Ниже показан простейший пример. Каждый элемент должен начинаться с включения

исходного корневого каталога, каковым в этом примере является `/home/duchess`, и заканчиваться его исключением:

```
# файл со списком для исключения
#
# включить домашний каталог
+ /duchess/
#
# включить .config и .local, исключить все остальные файлы с именами,
# начинающимися с точки
+ /duchess/.config
+ /duchess/.local
- /duchess/.*
#
# включить jazz/, исключить все остальные файлы в Documents
+ /duchess/Documents/
+ /duchess/Documents/compositions/
+ /duchess/Documents/compositions/jazz/
- /duchess/Documents/compositions/*
- /duchess/Documents/*
#
# включить schedule.odt, исключить все файлы .ogg
# в arias/, исключить все остальные файлы в Music
+ /duchess/Music/
+ /duchess/Music/schedule.odt
+ /duchess/Music/arias/*.ogg
- /duchess/Music/arias/*
- /duchess/Music/*
#
# включить courses/, исключить все остальные файлы в Videos
+ /duchess/Videos/
+ /duchess/Videos/courses/
- /duchess/Videos/*
#
# исключить все остальные файлы
- /duchess/*
```

Файл с таким списком нужно передать команде `rsync` в параметре `--exclude-from=`:

```
duchess@pc:~$ rsync -av ~ \
  --exclude-from=/home/duchess/exclude-list.txt \
  /media/duchess/2tbdisk/
```

Комментарий

С файлом `exclude-list.txt` команда `rsync` скопирует:

- два файла: `.config` и `.local`;
- подкаталог `/jazz` в `/Documents`;

- файл `schedule.odt` из `/Music` и файлы `.ogg` из `/Music/arias/`;
- каталог `/courses` из `/Videos`.

В файле со списком не должно быть пустых строк, а комментарии (#) помогают оставлять себе напоминания о цели каждого раздела и служат для разделения блоков строк. Пути к включаемым элементам должны начинаться со знака плюс, а исключаемые — со знака минус.

Все остальные файлы в `/home/duchess` исключаются из резервного копирования. Включаемые элементы всегда должны следовать первыми. Как показано в примере файла, необходимо быть максимально точными при определении каждого включения/исключения. Включаемые объекты должны перечисляться в порядке иерархии их каталогов со всеми подкаталогами. Например, используя такой список, вы не получите ожидаемого результата:

```
+ /duchess/Documents/compositions/
- /duchess/*
```

Можно попробовать включить `/Documents`:

```
+ /duchess/Documents/
+ /duchess/Documents/compositions/
- /duchess/*
```

Но теперь копироваться будут все подкаталоги в `/Documents` с их содержимым, а не только `/compositions`. Чтобы скопировать только `/compositions`, мало исключить только `/duchess`, нужно также исключить `/Documents`. Следующий пример скопирует только `/duchess/Documents/compositions/` и ничего больше:

```
+ /duchess/Documents/
+ /duchess/Documents/compositions/
- /duchess/Documents/*
- /duchess/*
```

Включать и исключать файлы по типам можно с помощью подстановочных знаков. Например, включить все файлы `.ogg` и `.flac`, исключить все файлы `.wav` и исключить все каталоги `cache` и `temp` можно следующим образом:

```
# включить домашний каталог
+ /duchess/
#
# включить все файлы ogg и flac
+ *.ogg
+ *.flac
#
# исключить файлы wav files, все каталоги cache и temp
```

```
- *.wav
- cache*
- temp*
```

В списке можно перечислить несколько исходных каталогов. Но целевой каталог может быть только один.

Дополнительная информация

- `man 1 rsync`
- Рецепт 7.10.

7.12. Ограничение скорости передачи в команде rsync

Задача

Передача больших файлов может отрицательно сказаться на работе других сетевых приложений. Нужен простой способ ограничить скорость передачи в `rsync` без использования сложных приемов и инструментов, например формирования трафика.

Решение

Используйте `rsync` с параметром `--bwlimit`. Следующий пример ограничит скорость передачи до 512 Кбайт/с:

```
$ rsync --bwlimit=512 -ave ssh ~/Music/arias empress@laptop:songs/
```

Комментарий

Параметр `--bwlimit` интерпретирует полученное значение только как килобайты в секунду.

Дополнительная информация

- `man 1 rsync`

7.13. Создание сервера резервного копирования с помощью rsyncd

Задача

Организовать для пользователей возможность резервного копирования их данных на центральном сервере, но без создания учетных записей для них на этом сервере.

Решение

Настройте центральный сервер резервного копирования и запустите `rsync` в режиме демона. У вас должна быть уже настроена служба имен, а хосты в вашей сети должны иметь доступ к серверу резервного копирования. В этом примере пользователям не потребуются учетные записи на сервере, поскольку управление доступом и авторизация будут осуществляться с помощью собственных средств `rsync`.



Только для локальной сети

Этот способ подходит только для использования в локальной сети, поскольку демон rsync не шифрует трафик при выполнении аутентификации или передаче файлов. Для шифрования трафика вам понадобится OpenVPN (глава 13).

`rsync` должен быть установлен на все машины. На сервере резервного копирования работает `rsyncd`, а клиенты используют команду `rsync` для подключения к серверу.

На сервере резервного копирования откройте в редакторе или создайте файл `/etc/rsyncd.conf` и добавьте следующее определение модуля архива:

```
# модули
[backup_dir1]
   path = /backups
   comment = "server1 public archive"
   list = yes
   read only = no
   use chroot = no
   uid = 0
   gid = 0
```

Создайте каталог `/backups` с набором разрешений 0700, принадлежащий пользователю root, чтобы предотвратить несанкционированный доступ со стороны любого, у кого есть доступ к серверу:

```
$ sudo mkdir /backups/
$ sudo chmod 0700 /backups/
```

Запустите `rsyncd` на сервере в режиме демона с помощью systemd:

```
$ sudo systemctl start rsyncd.service
```

В Debian/Ubuntu используйте `rsync.service`.

Если в вашем дистрибутиве Linux отсутствует systemd, то выполните простую команду `rsync`:

```
admin@server1:~$ sudo rsync --daemon
```

На сервере резервного копирования проверьте, принимает ли соединения rsyncd:

```
admin@server1:~$ rsync server1::
backup_dir1     "server1 public archive"
```

Затем выполните ту же проверку с другого компьютера в вашей сети, указав имя хоста или IP-адрес сервера:

```
duchess@pc:~$ rsync server1::
backup_dir1 "server1 public archive"

duchess@pc:~$ rsync 192.168.10.15::
backup_dir1 "server1 public archive"
```

Если все в порядке, то можно приступать к копированию файлов. Проверьте возможность копирования файлов на свой новый сервер rsyncd:

```
duchess@pc:~$ rsync -av ~/drawings server1::backup_dir1
building file list.....done
drawings/
drawings/aug_03
drawings/sept_03

wrote 1126399 bytes  read 104 bytes 1522.0 bytes/sec
total size is 1130228 speedup is 0.94
```

Затем полюбуйтесь на скопированные файлы:

```
duchess@pc:~$ rsync server1::backup_dir1/drawings/
drwx------    4,096  2021/01/04  06:06:55    .
-rw-r--r--   21,560  2021/09/17  08:53:18    aug_03
-rw-r--r--   21,560  2021/10/14  16:42:16    sept_03
```

Выгрузите еще несколько файлов на сервер, далее скачайте файлы на другой компьютер с сервера rsyncd:

```
madmax@buntu:~$ rsync -av server1::backup_dir1/drawings ~/downloads
receiving incremental file list
created directory /home/madmax/downloads
drawings/
drawings/aug_03
drawings/sept_03

sent 123 bytes received 11562479 bytes 1755.00 bytes/sec
total size is 1141776 speedup is 1.00
```

Все в порядке. Сделайте перерыв и насладитесь своей маленькой победой.

Комментарий

Это небезопасный способ передачи файлов, поскольку не поддерживает шифрования, и любой, кто подключен к сети, сможет заглянуть в файлы. Но в локальной сети это вполне удобное решение для архивирования и обмена файлами.

Двойное двоеточие в команде `rsync [`*`имя_хоста`*`]::` при подключении к серверу rsync, работающему в режиме демона, требует от rsync указанный модуль.

Рассмотрим поближе настройки в примере файла `/etc/rsyncd.conf`:

- `[backup_dir1]` — имя модуля, может быть любым по вашему выбору;
- `path =` — определяет каталог для использования модулем;
- `comment =` — краткое описание того, кому принадлежит модуль или для чего он предназначен;
- `list=yes` — позволяет пользователям видеть список файлов в модуле. Значение `no` скроет модуль;
- `read only = no` — позволяет пользователям выгружать файлы на сервер;
- `use chroot = no` — отменяет настройку по умолчанию `chroot = yes`. Утилиту chroot (change root — «сменить корень») иногда называют chroot jail — *«клетка chroot»*. Chroot jail — это отдельная область внутри файловой системы, которая содержит собственную корневую файловую систему, команды, библиотеки и все остальное, что необходимо для работы. Это небезопасная среда, хотя обычно ее причисляют к средствам безопасности. В руководстве man для `rsync` она описывается как полезная защита от ошибок в конфигурации. При использовании данного средства `rsync` не сможет проследовать по символическим ссылкам за пределы ограниченной области, и это услож-

няет сохранение UID и GID по имени. Но все субъективно, и вполне может быть, что для вас это хороший вариант. См. раздел об использовании chroot в руководстве: `man 5 rsyncd.conf`.

Установите для `uid` и `gid` значение `root` или `0`. Это обеспечит сохранность UID и GID и разрешений.

В случае неудачной попытки передать файлы посмотрите сообщения об ошибках `rsync`. Они помогут вам, если вы допустили ошибку в пути к файлам, неправильно написали что-то или не смогли подключиться к серверу, и дадут полезные советы по устранению проблемы.

Если вы используете Linux без systemd, то обратитесь к документации для этого дистрибутива, чтобы узнать, как запустить и остановить rsyncd.

См. рецепт 7.14, чтобы узнать, как настроить управление доступом.

Дополнительная информация

- `man 5 rsyncd.conf`

7.14. Ограничение доступа к модулям rsyncd

Задача

Для вас нежелательно использовать открытый сервер rsyncd, а также хотелось бы, чтобы пользователи имели свои защищенные модули, недоступные другим пользователям.

Решение

Сервер rsyncd поддерживает простую аутентификацию и имеет средства управления доступом. Создайте новый файл, содержащий пары «имя пользователя/пароль», и добавьте в файл `/etc/rsyncd.conf` директивы `auth users` и `secrets file`.

Сначала создайте файл паролей. В следующем примере показано содержимое файла `/etc/rsyncd-users` со списком из трех пользователей и их паролями:

```
# rsync-users для server1
duchess:12345
madmax:23456
stash:34567
```

Настройте для этого файла разрешения на чтение и запись только для root:

```
$ sudo chmod 0600 /etc/rsyncd-users
```

Определите в файле `/etc/rsyncd.conf` модули для каждого пользователя. В следующем примере показано определение модуля для Duchess, который выделяет для резервных копий этого пользователя каталог `/backups/duchess` на сервере rsync:

```
[duchess_backup]
   path = /backups/duchess
   comment = Duchess's private archive
   list = yes
   read only = no
   auth users = duchess
   secrets file =/etc/rsyncd-users
   use chroot = no
   strict modes = yes
   uid = root
   gid = root
```

Создайте объявленные в модулях каталоги с разрешениями 0700, как показано ниже на примере для Duchess:

```
$ sudo mkdir /backups/duchess/
$ sudo chmod -R 0700 /backups/duchess/
```

Попробуйте аутентифицироваться:

```
$ rsync duchess@server1::duchess_backup
Password: 12345
drwxr-xr-x 4,096 2020/06/29 18:24:43 .
```

И попытайтесь отправить несколько файлов:

```
$ rsync -av ~/logs duchess@server1::duchess_backup
Password:
sending incremental file list
logs/
logs/irc.log
logs/irc_#core-standup.log
logs/irc_#core.log
logs/irc_#desktop.log
logs/irc_#engineering.log
logs/irc_#mobile.log
```

```
sent 130,507 bytes received 305 bytes 37,374.86 bytes/sec
total size is 129,383 speedup is 0.99
```

Работает! Если попытка передать файлы потерпела неудачу, то загляните в журнал rsync и узнайте причину. В systemd Linux самые последние записи в журнале можно получить, запросив статус службы:

```
$ systemctl status rsyncd.service
```

В других дистрибутивах журнал демона rsyncd должен находиться в `/var/log`.

Комментарий

Пары «имя пользователя/пароль» могут быть совершенно произвольными, поскольку они никак не связаны с системными учетными записями пользователей. Пользователи rsyncd не имеют доступа к хост-системе за границами выделенных им каталогов.

Для большей безопасности добавьте в файл `/etc/rsyncd.conf` следующие директивы:

- `hosts allow` — ее можно использовать для перечисления хостов, которым разрешен доступ к архивам rsyncd. С ее помощью, например, можно разрешить доступ только хостам из одной подсети:

  ```
  hosts allow = *.local.net
  hosts allow = 192.168.1.
  ```

  Всем остальным хостам доступ будет закрыт, поэтому можно обойтись без директивы `hosts deny`;
- `hosts deny` — обычно нет нужды использовать эту директиву, если уже работает `hosts allow`. С ее помощью удобно закрыть доступ определенным хостам, вызывающим раздражение.

Файл паролей хранится в обычном текстовом виде, поэтому должен быть доступен только суперпользователю.

Дополнительная информация

- `man 5 rsyncd.conf`
- Подраздел «Комментарий» в рецепте 7.13, где рассказывается о параметрах команды.

7.15. Создание сообщения с приветствием для rsyncd

Задача

Вы настраиваете сервер rsyncd, и у вас появилась идея приветствовать пользователей забавными сообщениями.

Решение

Запишите свое сообщение в обычный текстовый файл, например в `/etc/rsync-motd`:

> *Добро пожаловать на ваш локальный сервер резервных копий! Не забудьте создать резервные копии своих файлов!*

Затем задайте местоположение файла в начале файла `/etc/rsyncd.conf`:

```
[global]
motd file = /etc/rsync-motd
```

Подключаясь к серверу, пользователи будут видеть ваше сообщение:

```
$ rsync server1::backup_dir1/
Добро пожаловать на ваш локальный сервер резервных копий!
Не забудьте создать резервные копии своих файлов!

drwx------          4,096 2020/06/29 18:24:43 .
-rwxr-xr-x          6,400 2015/03/13 08:21:21 keytool
drwx------          4,096 2020/06/17 06:07:41 WIP
drwx------          4,096 2020/06/17 06:06:55 bin
drwxr-xr-x          4,096 2020/06/30 09:47:42 duchess
[...]
```

Комментарий

Вывод приветственных сообщений — старая добрая традиция Unix. Эта возможность позволяет поприветствовать пользователей, объявить о простоях в обслуживании, дать советы по безопасности, резервному копированию и всему тому, что вы считаете важным.

Дополнительная информация

- `man 5 rsyncd.conf`

ГЛАВА 8

Управление дисковыми разделами с помощью parted

Все накопители большой емкости — жесткие диски SATA, твердотельные накопители, USB-накопители, карты SD (Secure Digital), NVMe (Non-Volatile Memory Express) и CompactFlash — должны быть разбиты на разделы и отформатированы, прежде чем их можно будет использовать. Все они поставляются с уже имеющимися разделами и файловыми системами, которые могут не соответствовать вашим целям. Кроме того, с течением времени цели могут меняться, и может потребоваться реорганизовать разделы на диске и/или использовать другие файловые системы. В этой главе вы узнаете, как можно управлять разделами с помощью команды `parted` (partition editor — редактор разделов).

Обзор

`parted` управляет только разделами, а о файловых системах рассказывается в главе 11. В главе 9 будет представлен графический интерфейс для `parted`, GParted, который поддерживает возможность управления не только разделами, но и файловыми системами.

Вы также узнаете о современной замене главной загрузочной записи (Master Boot Record, MBR) — устаревшей и неадекватной таблице разделов. На смену MBR пришла новая таблица глобальных уникальных идентификаторов разделов (Globally Unique Identifier Partition Table, GPT).

`parted` показывает информацию о разделах, а также может создавать, удалять и изменять размеры разделов. У `parted` есть только одна проблема: он

немедленно записывает все изменения на диск, поэтому нужно быть очень осторожными при обращении с этой утилитой. Графический интерфейс GParted удобнее в этом отношении — он не применяет изменения, пока вы не нажмете кнопку.

Все запоминающие устройства по привычке называют *дисками*, хотя многие из них не являются таковыми, как, например, твердотельные накопители. А почему бы и нет? Ведь мы до сих пор говорим: «позвони по телефону» или «сфотографируй», подразумевая использование смартфона.

Возможность деления дискового пространства на разделы позволяет создать одну или несколько логически изолированных областей. На диске должен иметься хотя бы один раздел. Количество разделов зависит только от ваших потребностей и капризов. После создания разделов в каждый из них нужно поместить файловую систему, и только тогда вы сможете использовать диск. На одном диске может быть несколько разделов, и каждый раздел может иметь свою файловую систему.

Имя диска в Linux всегда начинается с `/dev` (сокращенно от device — «устройство»). Например, `/dev/sda` — это имя жесткого диска, а `/dev/sr0` — привода оптических дисков. Имена разделов образуются из имени диска и номера. Если на диске `/dev/sda` имеется три раздела, то они будут называться `/dev/sda1`, `/dev/sda2` и `/dev/sda3`.

Схемы деления на разделы

Некоторые дистрибутивы Linux предлагают установить все в один раздел. Это вполне работоспособный вариант, однако создание еще нескольких разделов во время установки дает дополнительные преимущества:

- наличие отдельного раздела для `/boot` упрощает управление мультизагрузочными системами, поскольку загрузочные файлы не зависят от операционных систем, которые вы устанавливаете или удаляете;
- выделение `/home` в отдельный раздел обеспечивает изоляцию домашних каталогов пользователей от корневой файловой системы и позволяет заменить установленную систему Linux, не касаясь `/home`. Более того, `/home` можно даже разместить на отдельном диске;
- каталоги `/var` и `/tmp` могут переполняться из-за выхода процессов из-под контроля. Размещение этих каталогов в отдельных разделах предотвращает подобное отрицательное влияние на другие файловые системы;

- размещение файла подкачки в отдельном разделе позволяет организовать перевод компьютера в спящий режим с сохранением состояния на диске.

Дополнительную информацию о выборе схемы деления диска на разделы можно найти в главе 1.

Таблицы разделов: GPT и MBR

Таблица разделов GUID (GUID Partition Table, GPT), впервые появившаяся в 2010 году, — это современная замена устаревшей главной загрузочной записи (Master Boot Record, MBR), оставшейся в наследство от PC-DOS. Если до сих пор вы использовали только MBR, то вас ждет приятный сюрприз, поскольку GPT — это свидетельство существенного улучшения.

Главная загрузочная запись MBR была придумана для компьютеров IBM еще в прошлом тысячелетии, в начале 1980-х, в захватывающую эпоху десятимегабайтных жестких дисков. MBR занимает первые 512 байт первого сектора диска, предшествующего первому разделу, и содержит загрузчик и таблицу разделов. Загрузчик занимает 446 байт, таблица разделов — 64 байта, а оставшиеся 2 байта хранят сигнатуру.

Шестьдесят четыре байта — это не так много для хранения большого количества чего-либо, поэтому в MBR может храниться информация только о четырех основных разделах. Один первичный раздел может содержать расширенный раздел, который затем можно разделить на логические разделы. Linux поддерживает (теоретически) неограниченное количество логических разделов. Но даже с огромным количеством логических разделов MBR ограничивает максимальный размер диска 2,2 ТиБ, чего в наши дни едва хватает для хранения мемов с котиками. Почему именно это ограничение? Посчитайте сами: MBR ограничена 32-битными адресами и может адресовать 2^{32} блока (блоки и секторы мы обсудим чуть позже), поэтому размер диска с 512-байтными блоками ограничивается $2^{32} \times 512 = 2{,}199023256 \times 10^{12}$ байтами.

BIOS и UEFI

GPT — часть спецификации единого расширяемого микропрограммного интерфейса (Unified Extensible Firmware Interface, UEFI). UEFI заменяет базовую систему ввода/вывода компьютера, более известную как BIOS. На рис. 8.1 показан интерфейс старой BIOS, а на рис. 8.2 — современного UEFI с богатым набором функций, что делает его похожим на маленькую операционную систему.

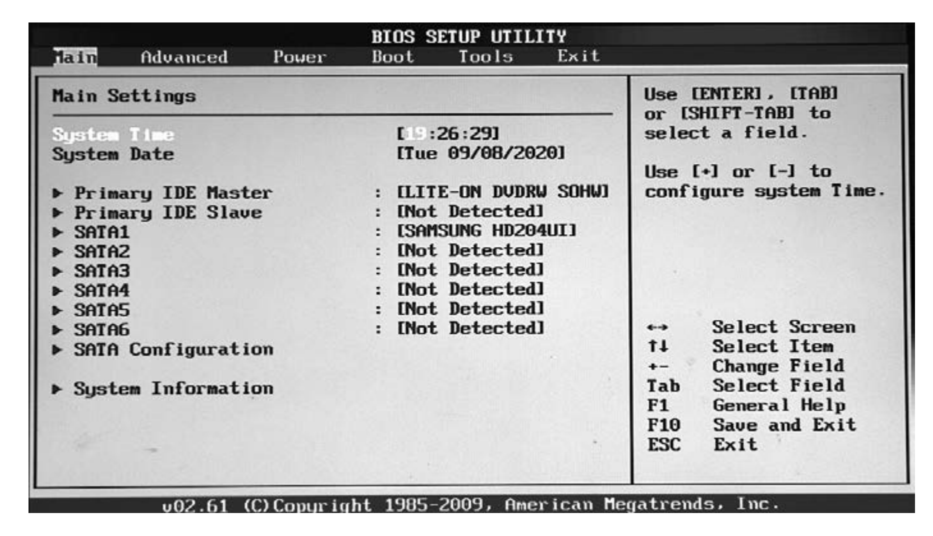


Рис. 8.1. Настройки устаревшей BIOS



Рис. 8.2. Настройки UEFI

GPT имеет множество преимуществ перед MBR:

- до 128 разделов в Linux с номерами 1–128 без всяких проблем с первичными и расширенными разделами;
- высокую отказоустойчивость: копии таблицы разделов хранятся в нескольких местах;

- уникальные идентификаторы дисков и разделов;
- поддержку устаревшего режима загрузки BIOS/MBR;
- проверку собственной целостности и таблицы разделов;
- безопасную загрузку.

MBR постепенно выходит из употребления, и вам определенно следует использовать GPT. В GPT первый сектор диска зарезервирован для защитной MBR, которая поддерживает GPT на компьютере с BIOS, благодаря чему имеется возможность применять GPT в старых системах с BIOS вместо UEFI. И загрузчик, и операционная система должны поддерживать GPT, и такая поддержка уже давно реализована в Linux. Единственная причина использования MBR — старые компьютеры со старыми операционными системами, которые не поддерживают GPT.

Если у вас есть подобная старая система, то вы не сможете обновить ее до поддержки UEFI — чтобы получить UEFI, вы должны будете заменить материнскую плату, поскольку и UEFI, и BIOS интегрированы в нее.

Блоки и секторы

Теперь поговорим о блоках и секторах и о том, как они влияют на максимальные размеры дисков, файлов и разделов. *Блоки* — это наименьшие единицы хранения на диске, которые может использовать файловая система. Деление на блоки — логическое, а не физическое. Самая маленькая физическая единица хранения — это *сектор*. Блоки могут занимать несколько секторов, а файл — несколько блоков.

При размещении файлов в блоках возникают некоторые потери, поскольку размеры файлов редко бывают кратными размерам блоков. Например, файл размером на один байт больше размеров четырех блоков займет пять блоков. Пятый блок будет хранить единственный байт и принадлежать только этому файлу. Из-за этого можно подумать, что 512-байтные блоки менее расточительны. Но в каждом блоке хранится еще кое-какая информация, помимо данных из файла.

Каждый блок, кроме данных из файла, хранит отметки времени, имя файла, атрибуты владения, разрешения, идентификатор блока и его правильный порядок с другими блоками, индексный узел и другие метаданные.

Блоки размером 4096 байт в восемь раз больше 512-байтных блоков, а метаданных хранят столько же. Чтобы полностью занять жесткий диск емкостью 4 ТиБ, на нем нужно разместить 8 000 000 000 блоков по 512 байт. При размере блока 4096 байт разместить нужно только 1 000 000 000 блоков — получается довольно внушительная экономия на метаданных.

Размер сектора ограничивает размер томов. Долгие годы стандартным считался размер сектора 512 байт, в настоящее время стандартным считается размер 4096 байт, поскольку емкость жестких дисков значительно выросла.

Таблица разделов GPT обеспечивает 64-битную адресацию, позволяя разместить 2^{64} блоков на одном диске, то есть размер жесткого диска с 512-байтными блоками может достигать 9 Збайт (9×10^{21} байт). С 4096-байтными блоками максимальный размер диска составляет 75 Збайт (75×10^{21} байт), чего, как мне кажется, будет достаточно даже для самого преданного коллекционера мемов с котиками. Это теоретические максимумы, но есть еще ограничения, накладываемые оборудованием, операционной системой и поддержкой больших томов в файловой системе. Например, файловая система Ext4 не может иметь размер больше 1 ЭиБ (1×2^{60}), а максимальный размер файла с размером блока 4096 байт составляет в ней 16 ТиБ. Файловая система XFS имеет максимальный размер 8 ЭиБ минус 1 байт ($8 \times 2^{60} - 1$).

Оптические CD и DVD имеют секторы размером по 2048 байт. Твердотельные устройства, такие как USB-накопители, SD-карты, CompactFlash и твердотельные накопители (SSD), тоже имеют секторы и блоки. Самая маленькая единица хранения на SSD называется *страницей*. Типичные размеры страниц — 2, 4, 8 Кбайт и больше. Блоки содержат от 128 до 256 страниц и обычно имеют размер от 256 Кбайт до 4 Мбайт.

От всех этих непривычных чисел голова может пойти кругом. В табл. 8.1 перечислены десятичные и двоичные единицы измерения, которые можно использовать для измерения емкости диска.

Таблица 8.1. Десятичные и двоичные единицы счисления

Величина	Десятичная единица	Величина	Двоичная единица
1	Б, байт	1	Б, байт
1000	Кбайт, килобайт	1024	КиБ, кибибайт
1000^2	Мбайт, мегабайт	1024^2	МиБ, мибибайт
1000^3	Гбайт, гигабайт	1024^3	ГиБ, гибибайт
1000^4	Тбайт, терабайт	1024^4	ТиБ, тебибайт
1000^5	Пбайт, петабайт	1024^5	ПиБ, пебибайт
1000^6	Эбайт, эксабайт	1024^6	ЭиБ, эксбибайт
1000^7	Збайт, зеттабайт	1024^7	ЗиБ, зебибайт
1000^8	Ибайт, йоттабайт	1024^8	ЙиБ, йобибайт

Десятичные единицы измерения — это степени 10; например, килобайт равен 1000 байт, или 10^3. Двоичные единицы — степени двойки, поэтому кибибайт равен 2^{10}, или 1024 байт. Производители жестких дисков любят использовать десятичный формат, чтобы емкость их дисков казалась больше.

Тот, кто придумал странную схему именования «бибайт», почти гарантировал, что никто и никогда не захочет произносить эти названия. В любом случае данное разделение вносит путаницу, поскольку многие используют эти именования единиц как взаимозаменяемые. Но, как бы то ни было, теперь вы знаете разницу.

8.1. Размонтирование разделов перед разбиением с помощью parted

Задача

Вы знаете, что перед изменением разделов с помощью `parted` их нужно размонтировать, но не знаете, как это сделать.

Решение

Размонтировать раздел можно в диспетчере файлов с графическим интерфейсом или воспользоваться командой `umount`. Ниже представлен пример размонтирования `/dev/sdc2`:

```
$ sudo umount /dev/sdc2
```

Но как узнать имя устройства? Загляните в рецепт 8.3, в котором рассказывается, как получить список всех подключенных дисков и разделов.

Если вы собираетесь создать новую таблицу разделов на диске, то должны размонтировать все разделы, имеющиеся на нем.



Внесение изменений в работающей системе

Весьма рискованно размонтировать файловые системы, такие как /home, /var или /tmp, размещенные в разных разделах и подключенные к активной корневой файловой системе. Безопаснее выполнять операции с разделами из другого экземпляра Linux, такого как SystemRescue (см. главу 19), или из второй системы Linux на той же машине (см. главу 1).

Комментарий

Технически монтируются и размонтируются файловые системы, а не разделы. Но я не против, если вы скажете «разделы».

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).

8.2. Запуск parted в командном режиме

Задача

Вы знаете, что команда `parted` поддерживает интерактивный режим, в котором можно работать как с командной оболочкой, и пакетный режим, когда `parted` выполняется как обычная команда. Теперь вы хотите узнать, как использовать оба режима.

Решение

Если ввести команду `parted` без параметров, то она запустит интерактивную оболочку `parted`. Для этого нужны привилегии root:

```
$ sudo parted
GNU Parted 3.2
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

Когда привычное приглашение к вводу сменится на `(parted)`, это будет означать, что вы находитесь в интерактивной оболочке `parted`. Введите **`help`**, чтобы получить список команд и их описания. Кроме того, можно получить справку для отдельных команд `parted`, например `help print`. Введите `quit`, чтобы выйти из `parted`. Большинство команд `parted` можно сократить до первой буквы, например `h` и `q`.

Чтобы запустить `parted` как обычную команду, нужно передать ей все необходимые параметры, как в следующем примере, который выводит список всех подключенных дисков:

```
$ sudo parted /dev/sdb print devices
/dev/sdb (2000GB)
/dev/sda (4001GB)
/dev/sdc (4010MB)
/dev/sdd (15.7GB)
/dev/sr0 (425MB)
```

Команда запустится, выполнит затребованное действие и вернется в обычную командную оболочку.

Комментарий

Будьте осторожны, задействуя любой режим, поскольку `parted` применяет изменения немедленно. Всегда создавайте резервные копии, прежде чем приступать к использованию `parted`.

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).

8.3. Обзор существующих дисков и разделов

Задача

Получить список имеющихся разделов, их размеры и типы файловых систем на них.

Решение

Если вам неизвестны имена дисков в вашей системе, то просто запустите `parted` без параметров:

```
$ sudo parted
GNU Parted 3.2
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

Если вы не указали устройство, то `parted` попытается угадать, какое вы хотите использовать — обычно первое, — и сообщит о своем выборе (обратите внимание на строку `Using /dev/sda` в предыдущем примере).

Запустите команду `print devices`, чтобы получить список дисков с их именами и размерами:

```
(parted) print devices
/dev/sda (256GB)
/dev/sdb (1000GB)
/dev/sdc (4010MB)
```

Выберите нужный вам диск, чтобы получить подробную информацию о нем:

```
(parted) select /dev/sdb
Using /dev/sdb
(parted) print
Model: ATA ST1000DM003-1SB1 (scsi)
Disk /dev/sdb: 1000GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system     Name  Flags
 1      1049kB  525MB   524MB   fat16                 boot, esp
 2      525MB   344GB   343GB   btrfs
 3      344GB   998GB   654GB   xfs
 4      998GB   1000GB  2148MB  linux-swap(v1)        swap

(parted)
```

Введите команду `quit`, чтобы выйти.

Можно открыть интерактивную оболочку `parted` для конкретного диска:

```
$ sudo parted /dev/sda
GNU Parted 3.2
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
```

Введите команду `print` без параметров, чтобы получить информацию об этом диске:

```
(parted) print
 Model: ATA SAMSUNG SSD SM87 (scsi)
Disk /dev/sda: 256GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
[...]
```

Команда `print all` позволяет вывести информацию обо всех разделах на всех устройствах:

```
(parted) print all
 Model: ATA SAMSUNG SSD SM87 (scsi)
```

```
Disk /dev/sda: 256GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End    Size    File system  Name                  Flags
 1      1049kB  524MB  523MB   fat16        EFI system            legacy_boot,
                                            partition             msftdata
 2      524MB   659MB  134MB                Microsoft reserved    msftres
                                            partition
 3      659MB   253GB  253GB   ntfs         Basic data partition  msftdata
 4      253GB   256GB  2561MB  ntfs                               diag

Model: ATA ST1000DM003-1SB1 (scsi)
Disk /dev/sdb: 1000GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system     Name  Flags
 1      1049kB  525MB   524MB   fat16                 boot, esp
 2      525MB   344GB   343GB   btrfs
 3      344GB   998GB   654GB   xfs
 4      998GB   1000GB  2148MB  linux-swap(v1)        swap

Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:
Number  Start   End     Size    Type     File system  Flags
 1      1049kB  4010MB  4009MB  primary  fat32
```

А так можно определить наличие на диске свободного пространства, не занятого ни одним разделом:

```
(parted) print free
Model: ATA ST4000DM000-1F21 (scsi)
Disk /dev/sda: 4001GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system     Name  Flags
        17.4kB  1049kB  1031kB  Free Space
 1      1049kB  500MB   499MB   ext4
 2      500MB   60.5GB  60.0GB  ext4
 3      60.5GB  2061GB  2000GB  xfs
 4      2061GB  2069GB  8000MB  linux-swap(v1)
        2069GB  4001GB  1932GB  Free Space
```

Комментарий

Разберем, что означает вся эта выводимая информация.

- `Model` (модель) — название производителя устройства.
- `Disk` (диск) — имя устройства и размер.
- `Sector size` (размер сектора) — физический и логический размеры блока. Логический размер блока 512 байт обеспечивает обратную совместимость со старыми дисковыми контроллерами и программным обеспечением.
- `Partition table` (таблица разделов) — сообщает тип таблицы разделов: msdos или gpt.
- `Flags` (флаги) — более важны для Windows, чем для Linux. Они идентифицируют типы разделов и в некоторых случаях необходимы, чтобы Windows не запуталась в них. Полный список флагов можно найти в руководстве пользователя Parted User's Manual (https://oreil.ly/SNyLL).

Ниже представлены несколько флагов с их описанием для примера:

- `legacy_boot` — отмечает загрузочный раздел GPT;
- `msftdata` — отмечает разделы GPT, содержащие файловые системы Microsoft: NTFS или FAT;
- `msftres` — раздел, зарезервированный Microsoft. Это специальный раздел, который необходим операционным системам Microsoft для использования разделов GPT. На дисках размером менее 16 Гбайт для зарезервированного раздела (MSR) отводится 32 Мбайт, а на дисках большего размера — 128 Мбайт.
- `diag` — раздел для восстановления Windows.
- `boot`, `esp` — оба флага отмечают раздел как загрузочный; `boot` — это метка для MBR, а `esp` — для GPT.
- `swap` — отмечает раздел подкачки.

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).

8.4. Создание разделов GPT на незагрузочном диске

Задача

Заново разбить диск на разделы, удалив все данные и создав новую таблицу разделов GPT. Это не загрузочный диск с операционной системой — он предназначен только для хранения данных.

Решение

Сначала создайте новую таблицу разделов, потом разделы, а затем убедитесь, что все они созданы правильно. Проверьте правильность выбора диска; см. рецепт 8.3, в котором рассказывается, как получить список дисков и разделов.

Следующий пример демонстрирует разбиение USB-накопителя `/dev/sdc`, который используется для хранения данных. Это не загрузочный диск с операционной системой. Перед запуском `parted` обязательно размонтируйте устройство. Первый шаг — размонтировать устройство, и только потом можно приступать к созданию новой таблицы разделов GPT:

```
$ sudo umount /dev/sdc
$ sudo parted /dev/sdc
GNU Parted 3.2
Using /dev/sdc
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) mklabel gpt
Warning: The existing disk label on /dev/sdc will be destroyed and all data on
this disk will be lost. Do you want to continue?
Yes/No? Yes
(parted) p
Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start  End  Size  File system  Name  Flags
```

Теперь можно создать новые разделы. Ниже приводится пример создания двух разделов с одинаковыми размерами. Для каждого раздела нужно указать имя, а также начало и конец:

```
(parted) mkpart "images" ext4 1MB 2004MB
(parted) mkpart "audio files" xfs 2005MB 100%
```

После этого проверьте получившийся результат и завершите сеанс работы с `parted`:

```
(parted) print
Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name         Flags
 1      1049kB  2005MB  2004MB  ext4         images
 2      2006MB  4009MB  2003MB  xfs          audio files

(parted) q
Information: You may need to update /etc/fstab.
```

Если начало или конец окажется слишком близко к другому разделу, то `parted` сообщит об ошибке. В следующем примере начало второго раздела совпадает с концом первого:

```
(parted) mkpart "images" ext4 2004MB 100%
Warning: You requested a partition from 2004MB to 4010MB (sectors
3914062..7831551).
The closest location we can manage is 2005MB to 4010MB (sectors
3915776..7831518).
Is this still acceptable to you?
Yes/No? Yes
```

Чтобы исправить ошибку, можно `2004MB` заменить на `2005MB`.

Комментарий

Синтаксис создания новых разделов GPT имеет вид: `mkpart name fs-type start end`.

Параметр `name` — обязательный. В нем можно передать любую строку, то есть вы можете дать разделу имя, которое будет напоминать вам его предназначение.

Метка `fs-type` — необязательная (тип файловой системы), но вы всегда должны указывать ее, чтобы разделу был присвоен правильный код типа файловой системы. Запустите команду `help mkpart` в интерактивной оболочке `parted`, чтобы увидеть список поддерживаемых меток файловой системы.

Параметр `start` определяет точку начала нового раздела. Это всегда числовое значение. Значение `1MB` в примере выше означает «отступить один мегабайт от

начала диска». Начать с нуля нельзя, поскольку первые 33 сектора зарезервированы для метки EFI, поэтому первый раздел может начинаться только с сектора с порядковым номером 34 или выше. Я использую отступ в один мегабайт, поскольку это легко запомнить.

В параметре `end` можно передать абсолютное или процентное значение размера. В предыдущем примере конец первого раздела находится на расстоянии 2005 Мбайт от начала диска. Второй раздел заканчивается в точке, соответствующей 100 % оставшегося пространства. Создание новой таблицы разделов удалит все данные, имевшиеся на диске.

Перед использованием нового раздела на него следует поместить файловую систему (см. главу 11).

Предупреждение You may need to update /etc/fstab (Возможно, вам следует поправить /etc/fstab) актуально, только если вы меняете разделы, перечисленные в файле `/etc/fstab`.

Даже если вы добавили в разделы метки с типами файловых систем, они не будут созданы автоматически. Создание файловых систем — это отдельный шаг.

Метки файловых систем иногда могут исчезнуть. Но если поместить файловую систему в раздел, то они сохранятся.

Справка в `parted` и документация не совсем четко описывают различия между созданием разделов GPT и MS-DOS. Создавая раздел GPT, вы должны придумать для него *имя*. При создании раздела MS-DOS нужно указать *тип раздела*: *первичный*, *расширенный* или *логический*. Это вызывает некоторую путаницу, и в результате администраторы создают разделы GPT с именами *«первичный»*, *«расширенный»* и *«логический»*. Это неверно, и для разделов GPT следует задавать *имена*.

В любом случае не следует создавать таблицы разделов MS-DOS, поскольку они устарели, разве только на старых компьютерах со старым программным обеспечением, которое не поддерживает GPT.

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).
- Глава 11.

8.5. Создание разделов для установки Linux

Задача

Определить, как разбить диск на разделы для установки Linux.

Решение

Используйте диспетчер разделов в программе установки Linux. Можно разбить диск на разделы и до запуска установки Linux, но использование диспетчера разделов в программе установки гарантирует, что все будет сделано правильно, и если вы допустите ошибку, то увидите соответствующее предупреждение. Предлагаемая схема разбиения приводится в рецепте 1.8.

Комментарий

Большинство программ установки Linux помогают разбить диск на разделы для новой установки, а также позволяют настроить разбиение вручную.

Дополнительная информация

- Вводная часть этой главы, где приводятся рекомендации по разбиению диска.
- Глава 1.

8.6. Удаление разделов

Задача

Удалить некоторые разделы.

Решение

Запустите `parted` в интерактивном режиме для диска, на котором требуется удалить разделы, и выведите таблицу разделов:

```
$ sudo parted /dev/sdc
GNU Parted 3.2
Using /dev/sdc
```

```
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p

Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:2

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  2005MB  2004MB  primary
 2      2005MB  4010MB  2005MB  primary
```

Чтобы, например, удалить второй раздел, введите команду `rm 2`. Раздел будет немедленно удален без подтверждения. Затем снова введите команду `p` для проверки:

```
(parted) rm 2
(parted) p
Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type     File system  Flags
 1      1049kB  2005MB  2004MB  primary
```

Комментарий

Будьте внимательны, указывая номер удаляемого раздела. Сделайте заметки на бумаге и не ленитесь перепроверить себя, прежде чем начать.

При попытке удалить смонтированный раздел `parted` выдаст предупреждение: Warning: Partition /dev/sdc2 is being used. Are you sure you want to continue? (Внимание: раздел /dev/sdc2 в настоящий момент используется. Вы уверены, что хотите продолжить?). Вы можете продолжить и удалить его. Любые открытые файлы останутся в памяти, пока вы не закроете их или не попытаетесь перезагрузить, но самое интересное то, что эти файлы можно сохранить в другом разделе.

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).

8.7. Восстановление удаленного раздела

Задача

Вы удалили раздел по ошибке и теперь хотите восстановить его.

Решение

Если вы случайно удалили новый пустой раздел, то нет смысла восстанавливать его, просто создайте его заново. Если в разделе имелась файловая система и данные, то приступить к восстановлению нужно немедленно. В интерактивной оболочке `parted` выполните команду `rescue` и укажите начало и конец раздела. Они могут быть приблизительными:

```
(parted) rescue 2000MB 4010MB
searching for file systems... 40%      (time left 00:01)Information: A ext4
primary partition was found at 2005MB -> 4010MB.  Do you want to add it to the
partition table?
Yes/No/Cancel? Yes
```

Оболочка `parted` не сообщает о результатах выполнения операции, поэтому выведите таблицу разделов, чтобы узнать, был ли восстановлен потерянный раздел:

```
(parted) p
Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name    Flags
 1      1049kB  2005MB  2004MB  xfs          images
 2      2005MB  4010MB  2005MB  ext4
```

Вот и все. Если вам повезет, то ваши файлы останутся в целости и сохранности.

Комментарий

Чем дольше вы раздумываете, прежде чем приступить к восстановлению раздела, тем ниже вероятность успешного восстановления, поскольку он может быть случайно перезаписан. Если нужно отложить спасательные операции на какое-то время, то спрячьте диск подальше в безопасное место.

Как всегда, не забывайте о резервном копировании.

Дополнительная информация

- `man 8 parted`
- Руководство пользователя Parted User's Manual (https://oreil.ly/SNyLL).

8.8. Увеличение размера раздела

Задача

Увеличить размер существующего раздела, в котором имеется файловая система.

Решение

Следующий пример демонстрирует, как увеличить размер раздела с файловой системой в нем. Для этого нужно выполнить два шага: изменить размер раздела, затем соответствующим образом изменить размер файловой системы. Каждая файловая система имеет свой набор инструментов, и вы должны использовать правильный инструмент для увеличения размера системы. В данном рецепте мы изменим размер разделов Ext4, XFS, Btrfs и FAT16/32.

Ext4, XFS и Btrfs можно увеличивать, не размонтируя их. Изменить размер FAT16/32 можно, только если эта файловая система размонтирована.

В конце раздела, который нужно увеличить, должно быть свободное место. Откройте интерактивную оболочку `parted` на выбранном диске и найдите свободное место:

```
$ sudo parted /dev/sdc
GNU Parted 3.2
Using /dev/sdc
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print free
Model: General USB Flash Disk (scsi)
Disk /dev/sdc: 4010MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name    Flags
[...]
        1024MB  2005MB  981MB   Free Space
 2      2005MB  3500MB  1495MB  ext4         audio
        3500MB  4010MB  510MB   Free Space
```

В данном случае мы имеем 981 Мбайт свободного пространства перед разделом 2 и 510 Мбайт — после него. Изменить можно только конечную точку раздела, поэтому следующие примеры расширяют раздел 2, добавляя к нему 510 Мбайт свободного места в конце.

Сначала расширим раздел до новой конечной точки:

```
(parted) resizepart 2 4010MB
```

Вы не увидите сообщения об успешном выполнении, но если была допущена ошибка, то сообщение об ошибке появится. Введите **p**, чтобы вывести таблицу разделов и убедиться, что команда `resizepart` сделала то, что вы хотели.

Теперь нужно расширить файловую систему, приведя ее размер в соответствие с новым размером раздела, использовав для этого соответствующую команду для данной файловой системы. В табл. 8.2 перечислены команды для каждой файловой системы, выполняющие их расширение для заполнения разделов.

Таблица 8.2. Команды для увеличения размеров файловых систем

Файловая система	Команда изменения размера
Ext4	sudo resize2fs /dev/sdc2
XFS	sudo xfs_growfs -d /dev/sdc2
Btrfs	sudo btrfs filesystem resize max /dev/sdc2
FAT16/32	sudo fatresize -i /dev/sdc2

Напомню еще раз, что перед изменением размера файловой системы FAT16/32 ее нужно размонтировать.

Выведите таблицу разделов в `parted`, чтобы проконтролировать результат.

Комментарий

Примеры в данной главе и в рецепте 8.9 небольшие, и в них фигурировал USB-накопитель емкостью 4 Гбайт. Он прекрасно подходит для тестирования, но в реальной жизни вы, скорее всего, будете использовать диски большего размера. Команды для работы с этими дисками останутся прежними, изменятся только размеры разделов.

Как всегда, перед началом работы обязательно убедитесь в наличии резервных копий.

Есть возможность уменьшить размер файловой системы, чтобы он был меньше размера раздела, но в этом нет особого смысла. Прочитайте главу 11, чтобы узнать все о создании файловых систем и управлении ими.

Если вам интересно, то я выбрала Ext4, Btrfs, XFS и FAT16/32, поскольку эти файловые системы чаще всего используются в Linux и хорошо поддерживаются.

Дополнительная информация

- Глава 11.
- `man 8 resize2fs`
- `man 8 parted`
- `man 8 xfs_growfs`
- `man 8 btrfs`
- `man 8 fsck.vfat`

8.9. Уменьшение размера раздела

Задача

Имеется раздел с файловой системой на нем, и его размер требуется уменьшить.

Решение

Размер файловой системы XFS нельзя уменьшить, можно только увеличить. Уменьшить можно Ext4, Btrfs и FAT16/32. Ext4 и FAT16/32 должны быть размонтированы перед сжатием. Btrfs можно сжать не размонтируя, но безопаснее сначала все-таки размонтировать ее.

Убедитесь, что используемая часть файловой системы, которую вы собираетесь сжать, меньше размера, до которого она будет сжиматься. Используйте команду `du`, чтобы узнать, сколько места занимают ваши файлы:

```
$ du -sh /media/duchess/shrinkme
922.6M    /media/duchess/shrinkme
```

Кроме того, около 40 % от суммарного объема файлов следует оставить для метаданных, не полностью занятых блоков и так, на всякий случай, поэтому в данном случае новый размер не должен быть меньше 1,4 Гбайт. Если понадобится место для добавления новых файлов, то учтите и это.

Уменьшать разделы немного сложнее, чем расширять. Процесс включает дополнительные шаги, и файловые системы должны размонтироваться перед сжатием. Если раздел находится на внешнем устройстве, таком как USB-накопитель, размонтируйте его и только потом приступайте к сжатию. Если раздел принадлежит вашей работающей системе, то, чтобы сжать его, следует загрузиться с аварийного диска или во второй экземпляр Linux в мультизагрузочной системе и запускать `parted` оттуда. Это необходимо для того, чтобы размонтировать файловую систему, которую требуется сжать.

После размонтирования выбранной файловой системы выполните следующие шаги:

- проверьте файловую систему;
- уменьшите размер файловой системы;
- уменьшите размер раздела.

Выполните следующую команду, чтобы проверить файловую систему Ext4:

```
$ sudo e2fsck -f /dev/sdc2
```

Чтобы проверить Btrfs:

```
$ sudo btrfs check /dev/sdc2
```

Чтобы проверить FAT16/32:

```
$ sudo fsck.vfat -v /dev/sdc2
```

По окончании проверки уменьшите размер файловой системы. Примеры команд в табл. 8.3 увеличивают размеры файловых систем до 2000 Мбайт.

Таблица 8.3. Команды для увеличения размеров файловых систем

Файловая система	Команда изменения размера
Ext4	sudo resize2fs /dev/sdc2 2g
Btrfs	sudo btrfs filesystem resize 2g /dev/sdc2
FAT16/32	sudo fatresize -s 2G /dev/sdc2

Теперь можно уменьшить размер раздела, приведя его в соответствие с размером файловой системы. Откройте интерактивную оболочку `parted` с выбранным устройством и запустите команду `resizepart`, указав номер раздела и конечную точку:

```
(parted) resizepart 1 2000MB
Warning: Shrinking a partition can cause data loss, are you sure you want
to continue?
Yes/No? y
```

Проверьте результат, запросив вывод таблицы разделов.

Комментарий

Устройства хранения постепенно дешевеют. В былые времена возня с разделами была необходима, чтобы втиснуть на диск большую часть файлов. Теперь у нас есть возможность настроить их размеры, как нам удобно.

Дополнительная информация

- Глава 11.
- `man 8 resize2fs`
- `man 8 parted`
- `man 8 btrfs`
- `man 8 fsck.vfat`

ГЛАВА 9

Управление разделами и файловыми системами с помощью GParted

GParted (GNOME Partition Manager) — диспетчер разделов для GNOME — один из моих любимых инструментов в Linux. GParted — это красивый графический интерфейс для `parted` и других команд управления файловыми системами. С помощью GParted можно создавать, удалять, перемещать, копировать и изменять размеры разделов и файловых систем, а также создавать новые таблицы разделов всего несколькими щелчками кнопкой мыши. Среди других возможностей можно также назвать восстановление данных и управление метками и UUID.

Метки разделов и файловых систем удобно использовать для их идентификации и присваивания файловым системам коротких и простых имен. Если метка не задана, то файловые системы идентифицируются своими длинными UUID. Например, если подключить USB-накопитель, не имеющий метки в файловой системе, то он будет доступен как `/media/username/1d742b2d-a621-4454-b4d3-469216a6f01e`. Присвойте ему красивую короткую метку, например `mystuff`, и тогда он будет монтироваться как `/media/username/mystuff`.



Внесение изменений в работающей системе

Для некоторых операций, таких как копирование, проверка и восстановление, а также установка меток и идентификаторов UUID, требуется сначала размонтировать файловые системы. ФС, необходимые работающей системе, размонтировать нельзя, поэтому используйте загрузочный компакт-диск или USB-накопитель SystemRescue (см. главу 19). Если вы работаете с мультизагрузочной системой с несколькими установленными системами Linux, то загрузитесь в другой экземпляр Linux и запустите GParted оттуда (см. главу 1).

Перед завершением работы GParted выводит окно состояния и предлагает сохранить файл журнала с перечислением выполненных действий. Сохраните эту информацию и изучите ее, поскольку по ней можно увидеть, какие команды и в какой последовательности вызывались.

Для работы GParted требуются привилегии root. При запуске GParted с привилегиями рядового пользователя откроется диалог для ввода пароля sudo или root (рис. 9.1).

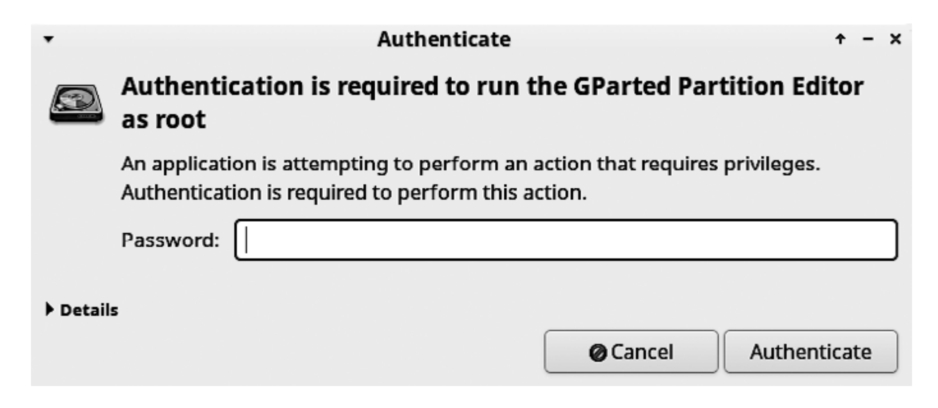


Рис. 9.1. Перед запуском приложения запрашивается пароль

Все запоминающие устройства по привычке называют *дисками*, даже твердотельные устройства, такие как SSD, USB-накопители, а также карты SD (Secure Digital), NVMe (Non-Volatile Memory Express) и CompactFlash. GParted может управлять любыми из этих дисков, внутренними и внешними, физически подключенными к системе.

Если вы не знакомы с основами разбиения дисков и управления файловыми системами, то прочитайте введение в главу 8.



Будьте осторожны!

Прежде чем попробовать любой из рецептов в этой главе, создайте резервные копии своих файлов и проверяйте правильность выбора дисков и разделов.

Создание новой таблицы разделов стирает все данные на диске.

Удаление или иное повреждение раздела приводит к потере всех данных в нем. Теоретически их можно восстановить, но это не гарантировано.

Для отработки практических навыков отлично подходят USB-накопители.

9.1. Обзор разделов, файловых систем и свободного пространства

Задача

Узнать, какие разделы и файловые системы имеются и сколько свободного места на подключенных дисках.

Решение

Запустите GParted и используйте раскрывающийся список вверху справа для переключения между дисками (рис. 9.2). Выберите пункт меню View ▸ Device Information (Вид ▸ Информация об устройстве), чтобы открыть слева панель с информацией о диске: названием модели, серийным номером, размером и типом таблицы разделов.

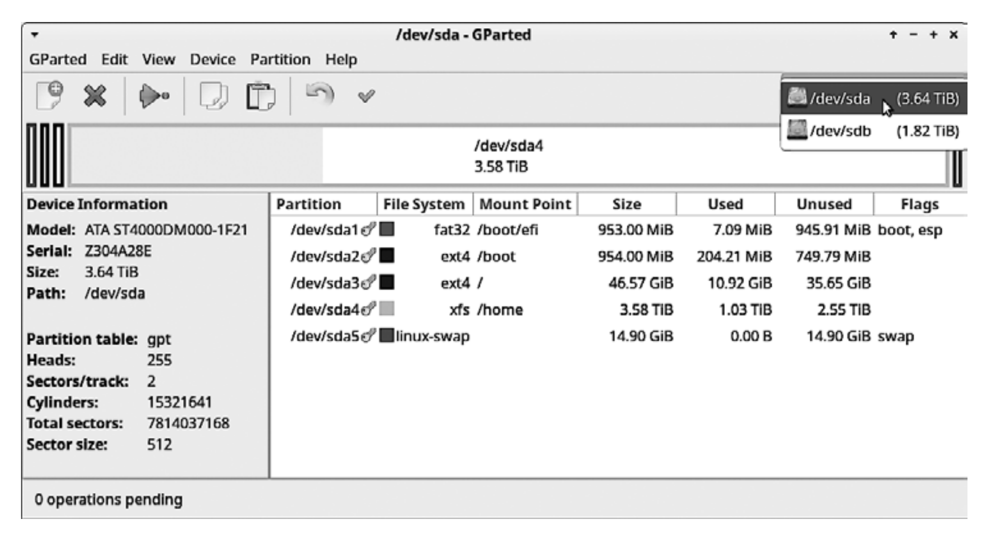


Рис. 9.2. Обзор дисков в GParted

Можно получить еще больше информации: имена устройств, точки монтирования, файловые системы, метки, типы и размеры разделов, занятое и общее пространство, а также свободное пространство. Щелкните правой кнопкой мыши на любом разделе, выберите в контекстном меню пункт Information (Информация), и на экране появится диалог с дополнительной информацией о выбранном разделе (рис. 9.3).

Комментарий

GParted не применяет изменений до тех пор, пока вы не нажмете кнопку с зеленым флажком на верхней панели инструментов, поэтому можете без опаски исследовать свои диски. Если вы случайно выберете какую-то команду, то для ее отмены нажмите кнопку с маленькой изогнутой желтой стрелкой рядом с флажком.

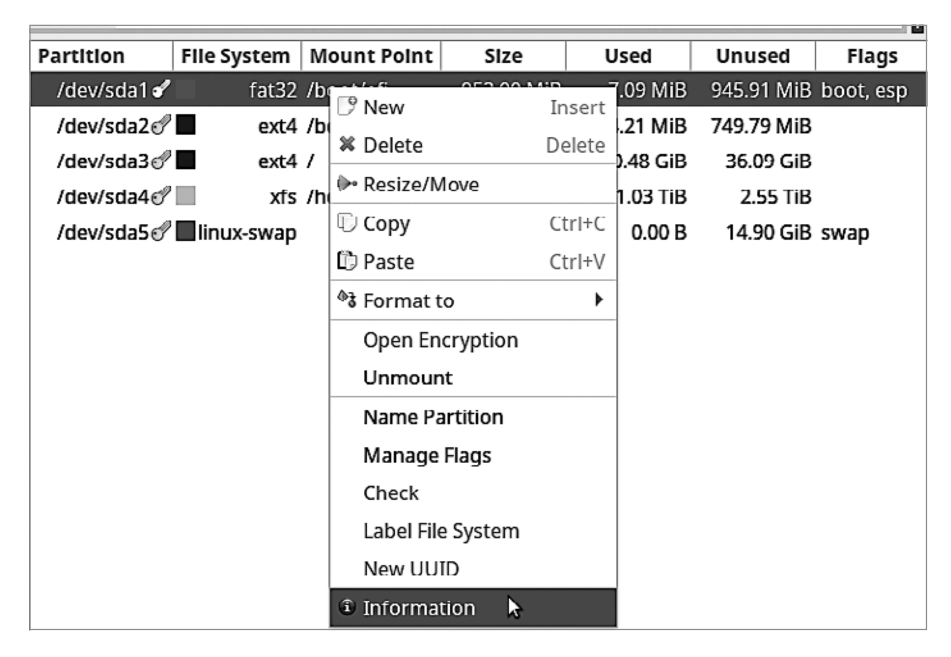


Рис. 9.3. Информация о разделе в GParted

В контекстном меню, открывающемся после щелчка правой кнопкой мыши на разделе, некоторые пункты неактивны, поскольку соответствующие им операции можно применять только к несмонтированным файловым системам. Выберите в контекстном меню пункт Unmount (Размонтировать), и эти команды станут доступны. Обратите внимание, что любые файловые системы, необходимые для текущей работающей системы, нельзя размонтировать; для манипуляций с ними используйте SystemRescue CD/USB (см. главу 19).

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).

9.2. Создание новой таблицы разделов

Задача

Реорганизовать диск и создать на нем новую таблицу разделов GPT. В настоящий момент на диске уже имеется таблица разделов MS-DOS, и ее нужно заменить на GPT. Возможно, это диск со старой ненужной информацией, которую не жаль потерять, и вы хотите начать с чистого диска.

Решение

Для начала проверьте и перепроверьте — на каком диске вы хотите создать новую таблицу разделов, поскольку, ошибившись, вы рискуете уничтожить нужные данные на другом диске. Это одна из операций, которую GParted применяет сразу после одного предупреждения, и отменить ее невозможно, так что будьте крайне осторожны.

Выберите диск в раскрывающемся списке вверху справа. Затем выберите пункт меню Device ▸ Create Partition Table (Устройство ▸ Создать таблицу разделов) (рис. 9.4).

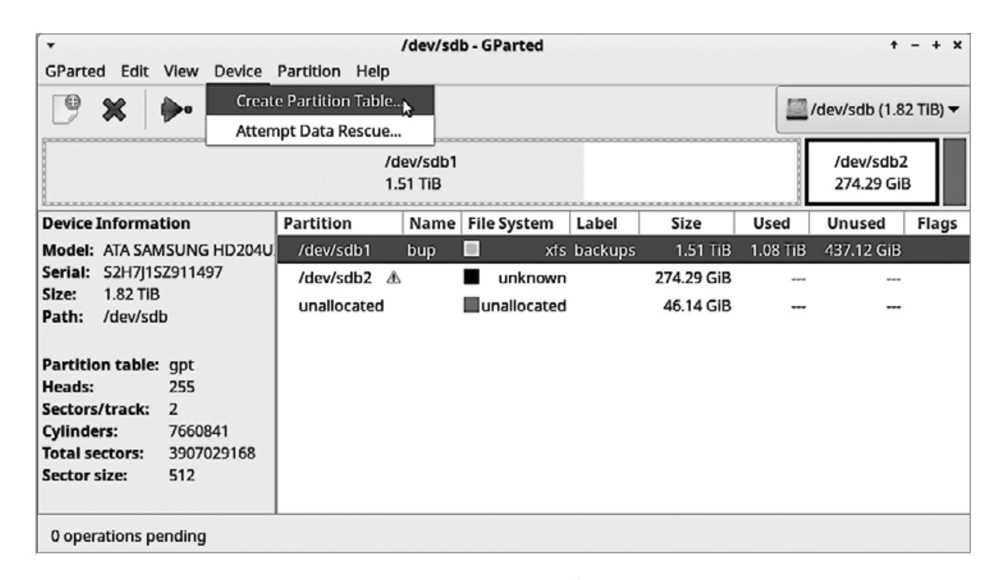


Рис. 9.4. Создание новой таблицы разделов

Выберите тип GPT таблицы разделов и нажмите кнопку Apply (Применить) (рис. 9.5).

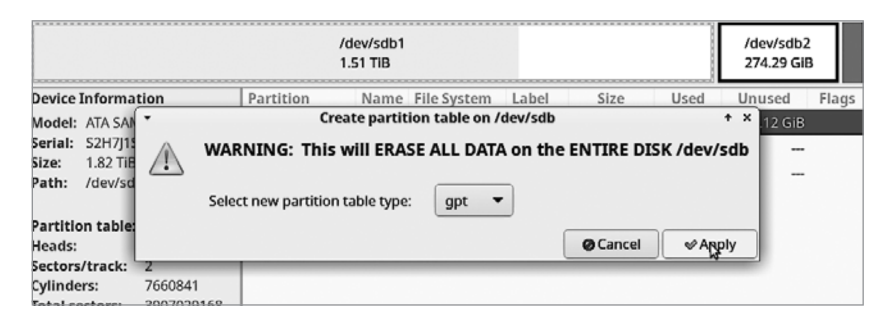


Рис. 9.5. Выбор типа таблицы разделов

Спустя несколько мгновений у вас будет новый, чистый, пустой диск, готовый к разбиению на разделы и форматированию с новыми файловыми системами.

Комментарий

Всегда выбирайте тип GPT для таблицы разделов, если нет веских причин использовать какой-то другой тип. GParted поддерживает несколько типов таблиц разделов, включая MS-DOS, BSD, Amiga и AIX. На платформе x86 чаще всего используются GPT и MS-DOS. Таблица GPT предназначена для современных больших жестких дисков, ею проще управлять и она устойчивее старой таблицы разделов MS-DOS. См. вводную часть в главе 8, где приводится подробная информация о таблицах разделов.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Глава 8.

9.3. Удаление раздела

Задача

Удалить один или несколько разделов.

Решение

Выберите раздел для удаления и щелкните на нем правой кнопкой мыши. Если в разделе есть смонтированная файловая система, то ее нужно сначала

размонтировать, выбрав в контекстном меню пункт Unmount (Размонтировать). После этого в том же меню выберите пункт Delete (Удалить), затем нажмите кнопку с зеленым флажком, и раздел исчезнет (рис. 9.6).

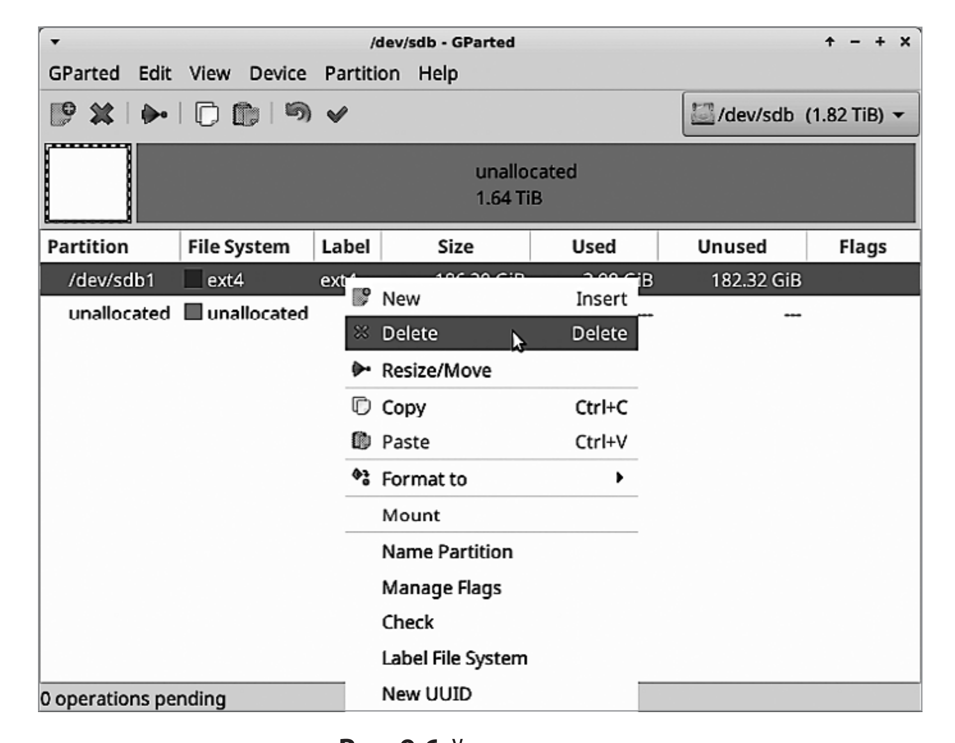


Рис. 9.6. Удаление раздела

По завершении удаления в строке состояния появится соответствующее сообщение.

Комментарий

Удаление раздела приведет к удалению всего, что находится внутри него, то есть если в разделе имеется файловая система с данными, то перед удалением убедитесь, что их не жалко потерять.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Рецепт 8.6.

9.4. Создание нового раздела

Задача

Создать новые разделы.

Решение

Для этого необходимо иметь на диске незанятое пространство. Следующий пример демонстрирует создание нового раздела размером 400 Гбайт и его форматирование с файловой системой Ext4 (рис. 9.7).

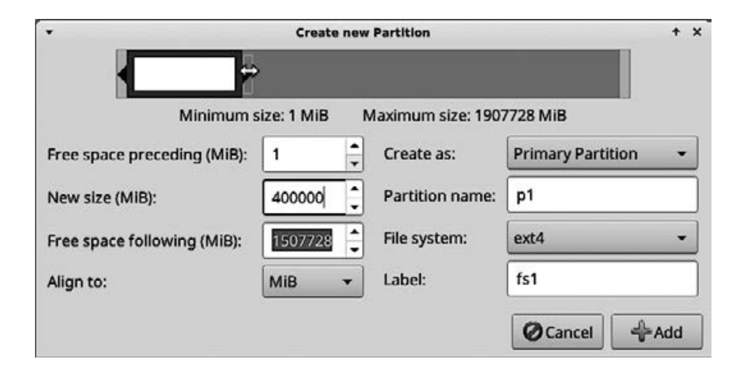


Рис. 9.7. Создание нового раздела

Выберите в меню пункт Partition ▸ New (Раздел ▸ Новый). После этого откроется диалог, где нужно ввести размер раздела, выбрать файловую систему и добавить метки для раздела и файловой системы. Для выбора размера используйте ползунок или поле ввода New Size (MiB) (Новый размер (МиБ)). Значения в поле New Size (Новый размер) измеряются в мибибайтах, то есть 400 000 МиБ — это 400 ГиБ. Затем нажмите кнопку Add (Добавить) в диалоге и кнопку с зеленым флажком на панели инструментов.

Закончив, обратитесь к главе 6, чтобы узнать, как правильно устанавливать права владения и разрешения в новой файловой системе.

Комментарий

При выборе таблицы разделов GPT вы всегда будете создавать только главные разделы. Два других варианта, логический и расширенный разделы, предназначены только для таблиц разделов MS-DOS. Если вы не помните тип таблицы

разделов на вашем диске, то выберите пункт меню View ▸ Device Information (Вид ▸ Информация об устройстве). Слева откроется панель с информацией о диске, включая тип таблицы разделов.

С помощью поля выбора типа файловой системы можно создать пустой раздел без файловой системы, если выбрать в нем значение Unformatted (Не отформатировано), находящееся в конце списка. Рядом с пунктом Unformatted (Не отформатировано) находится пункт Clear (Очистить), который удаляет существующую файловую систему и сохраняет раздел.

GParted объединяет создание раздела и размещение в нем файловой системы в одну быструю операцию. Это упрощает и ускоряет создание файловых систем, по сравнению с использованием команды `parted`, которая создает только разделы и требует, чтобы вы создавали файловые системы самостоятельно.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Глава 8.
- Глава 11.

9.5. Удаление файловой системы без удаления раздела

Задача

Удалить файловую систему, не удаляя раздел, в котором она находится, поскольку вы решили отформатировать раздел для использования в другой файловой системе или существующая файловая система была повреждена и вы хотите повторно отформатировать ее и затем скопировать файлы обратно в нее (рис. 9.8).

Решение

Файловую систему сначала нужно размонтировать. Щелкните на разделе правой кнопкой мыши и в контекстном меню выберите пункт Unmount (Размонтировать). По выполнении этой операции снова щелкните на разделе правой

кнопкой мыши, наведите указатель на пункт Format To (Форматировать в), прокрутите появившийся список до конца и выберите Cleared (Очищены). Эта команда удалит файловую систему, не удаляя раздел.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).

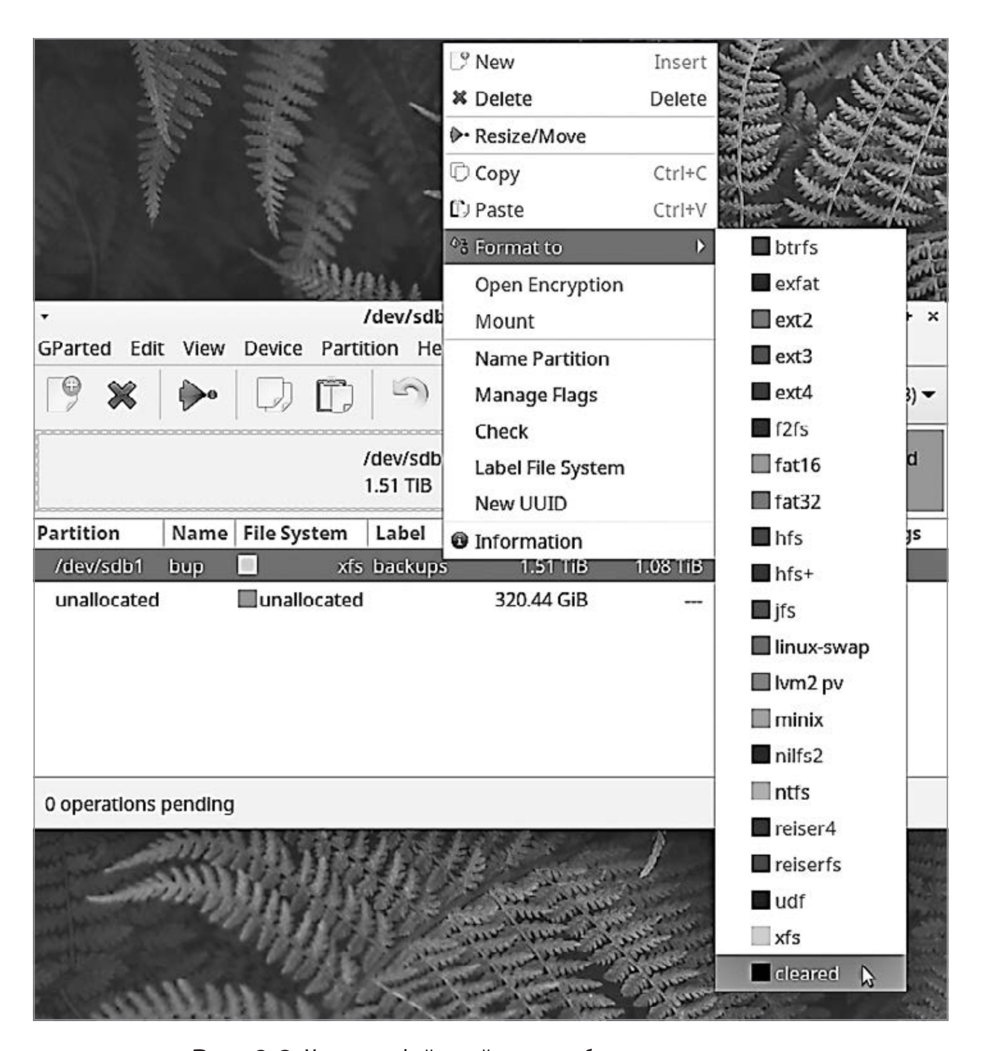


Рис. 9.8. Удаление файловой системы без удаления раздела

9.6. Восстановление удаленного раздела

Задача

Вы удалили раздел и теперь хотите восстановить его.

Решение

Если вы случайно удалили новый пустой раздел, то нет смысла пытаться восстанавливать его — просто создайте его снова. Если в удаленном разделе была файловая система и данные, то лучше приступить к восстановлению немедленно. Выберите в меню пункт Device ▸ Attempt Data Rescue (Устройство ▸ Попробовать восстановить данные).

Это может занять много времени, и нет никаких гарантий успеха. Команда `parted`, кажется, делает это быстрее; см. рецепт 8.7.

Комментарий

Обычно быстрее создать новый раздел и файловую систему, а затем восстановить файлы из резервной копии. Но попытка восстановить раздел, я думаю, не помешает.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Рецепт 8.7.

9.7. Изменение размера раздела

Задача

Увеличить или уменьшить раздел.

Решение

В GParted данная операция выполняется в несколько щелчков кнопкой мыши. При изменении размера раздела следует также изменить размер файловой системы. GParted объединяет эти две операции в одну.

Чтобы увеличить раздел, на диске должна существовать неразмеченная свободная область, расположенная сразу за разделом. Размеры файловых систем

Ext4, Btrfs и XFS можно изменять, не размонтируя их. Но файловую систему FAT16/32 обязательно нужно размонтировать.



Всегда создавайте резервные копии!

Не забывайте своевременно создавать резервные копии!

На рис. 9.9 показана файловая система FAT32, за которой следует большая неразмеченная область.

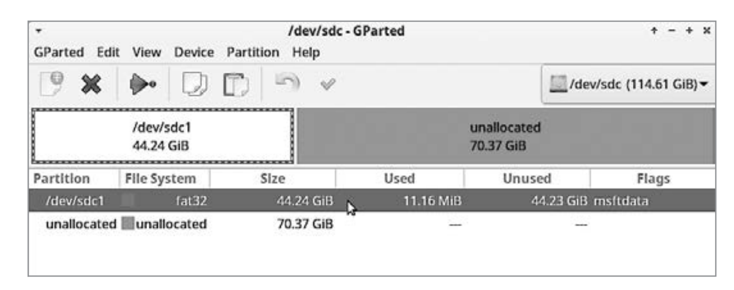


Рис. 9.9. Выбор раздела для изменения размера

Щелкните правой кнопкой на разделе и в контекстном меню выберите пункт Resize/Move (Изменить размер или переместить). В открывшемся диалоге вы сможете установить новый размер либо переместив ползунок, либо указав значение в мибибайтах в поле New Size (Новый размер) (рис. 9.10).

Рис. 9.10. Настройка нового размера раздела

Нажмите кнопку Resize/Move (Изменить размер или переместить) в диалоге и затем кнопку с зеленым флажком. Увеличение раздела обычно занимает 1–2 минуты, и по завершении появится сообщение в строке состояния.

Уменьшение размера раздела выполняется аналогично, за исключением того, что для этой операции не требуется наличие неразмеченной свободной области в конце. Новый размер раздела должен быть как минимум на 10 % больше пространства, занятого файлами. Даже если вы не планируете добавлять новые файлы в эту файловую систему, все равно оставьте некоторый свободный объем, поскольку после заполнения файловой системы до конца вы не сможете получить к ней доступ. Уменьшение раздела занимает больше времени, чем его увеличение.

Комментарий

Файловая система Ext4 резервирует небольшой объем для пользователя root. Даже если файловая система заполняется, то root все равно сможет получить доступ к ней и удалить ненужные файлы. FAT16/32, Btrfs и XFS не имеют зарезервированных блоков.

Ext4 и Btrfs можно сжать, не размонтируя их. XFS можно только увеличить, но нельзя сжать. Перед изменением размера их безопаснее размонтировать.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Рецепт 8.8.
- Рецепт 8.9.

9.8. Перемещение раздела

Задача

На диске есть немного свободного места между разделами, например между `/dev/sda1` и `/dev/sda2`. Вы хотите переместить `/dev/sda2` в свободное место, чтобы между ними не было промежутка. Или, напротив, увеличить раздел `/dev/sda1`, но после него нет свободного места, поэтому вам нужно переместить `/dev/sda2`, чтобы освободить пространство.

Решение

Щелкните правой кнопкой мыши на разделе, который требуется переместить, и в контекстном меню выберите пункт Resize/Move (Изменить размер или переместить) (рис. 9.11).

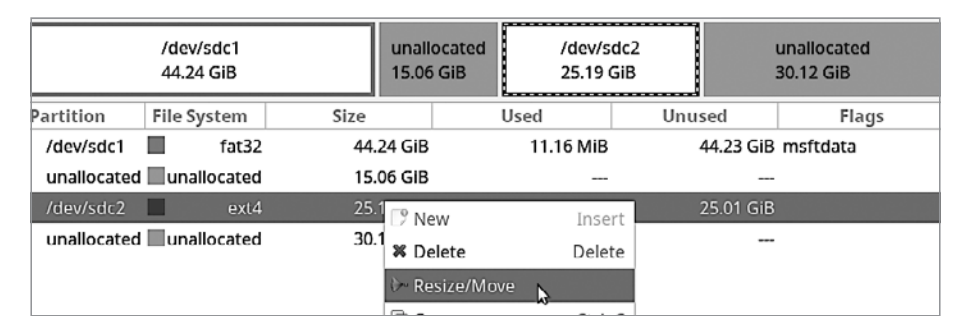


Рис. 9.11. Выбор раздела для перемещения

В открывшемся диалоге Resize/Move (Изменение размера) переместите ползунок влево или введите `0` в поле Free Space Preceding (MiB) (Свободное место до (МиБ)). Затем нажмите кнопку Resize/Move (Изменить размер или переместить) (рис. 9.12).

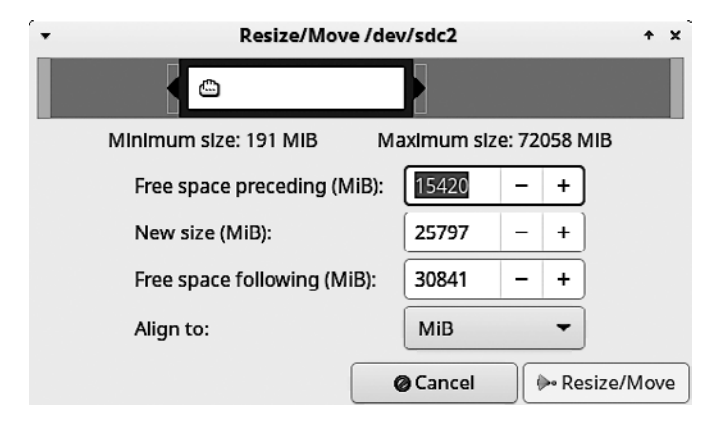


Рис. 9.12. Перемещение раздела

На выполнение операции может уйти довольно много времени, порой до нескольких часов, в зависимости от объема данных в разделе.

Комментарий

Переместить раздел сложнее, чем изменить его размер. При изменении размера раздела перемещается только его конечная точка, а при перемещении требуется также переместить его начальную точку, что для операционной системы является большим изменением. GParted обычно уверенно справляется с данной

операцией, и все же она остается довольно рискованной, поэтому всегда создавайте заранее резервные копии.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Глава 19.

9.9. Копирование раздела

Задача

Создать копию (клон) раздела или нескольких разделов, чтобы сохранить как резервную копию или перенести данные на новый жесткий диск.

Решение

Используйте операцию копирования. Например, чтобы создать копию раздела `/dev/sdb2` на жестком диске, подключенном к системе через USB, выделите на нем неразмеченную область, равную или большую, чем размер копируемого раздела.

Щелкните правой кнопкой на разделе, который нужно скопировать (рис. 9.13). Размонтируйте его, если он смонтирован, затем выберите в контекстном меню пункт Copy (Копировать).

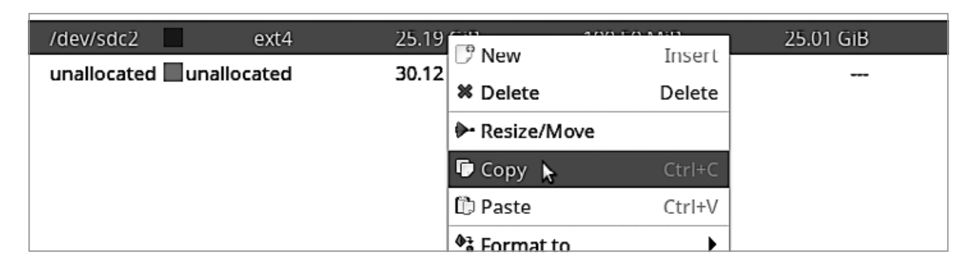


Рис. 9.13. Копирование раздела

Перейдите на диск, куда нужно скопировать выбранный раздел, выберите неразмеченную область, щелкните на ней правой кнопкой мыши и выберите пункт контекстного меню Paste (Вставить). Откроется диалог настройки параметров увеличения размера и изменения местоположения нового раздела (рис. 9.14). Выполнив настройки, нажмите Paste (Вставить).

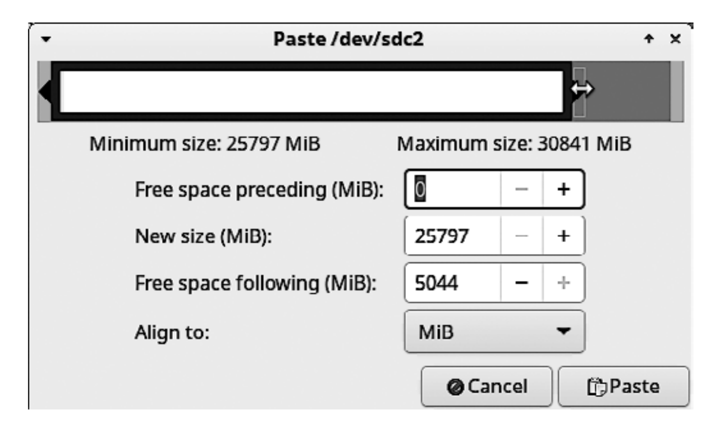


Рис. 9.14. Настройки создания нового раздела для копии

Чтобы начать копирование, нажмите кнопку с зеленым флажком. Если вы передумали, то нажмите кнопку отмены с желтой изогнутой стрелкой. Операция копирования займет немного времени, в зависимости от объема копируемых данных.

Комментарий

Содержимое выбранного раздела должно скопироваться в новый раздел такого же или большего размера. Копирование раздела в неразмеченное пространство избавляет от хлопот по созданию целевого раздела.

По своему опыту могу сказать, что копирование разделов имеет ограниченную ценность. Копируются не только данные, но и UUID раздела и файловой системы, скопированный раздел нельзя смонтировать в той же системе одновременно с исходным, не изменив UUID. (Впрочем, это легко сделать в GParted с помощью контекстного меню.) После изменения UUID файловых систем, перечисленных в `/etc/fstab`, соответствующие им записи нужно отредактировать. Я думаю, что в большинстве случаев проще создавать новые разделы и файловые системы, а затем копировать в них свои файлы.

Дополнительная информация

- Главная страница GNOME Partition Editor (https://gparted.org).
- Глава 11.
- Рецепт 11.6.
- Глава 19.

9.10. Управление файловыми системами с помощью GParted

Задача

Нужен хороший инструмент с графическим интерфейсом для создания новых файловых систем.

Решение

Используйте GParted, который может управлять разделами и файловыми системами. Выберите раздел, который нужно отформатировать в новую файловую систему, щелкните на нем правой кнопкой мыши и выберите нужную файловую систему (рис. 9.15).

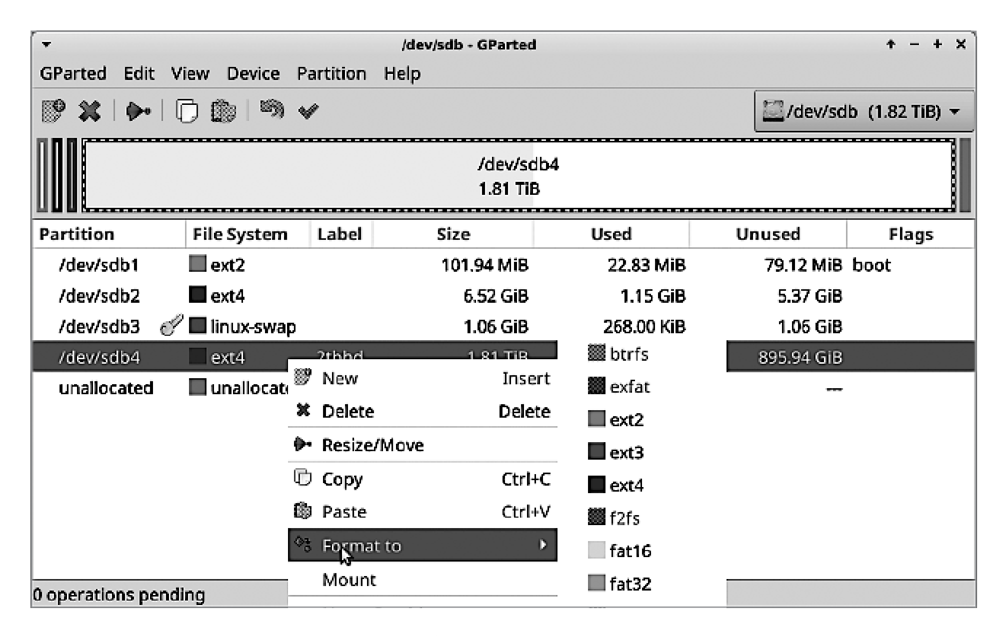


Рис. 9.15. Выбор файловой системы в GParted

Затем нажмите кнопку с зеленым флажком на панели инструментов, чтобы создать новую файловую систему. Имейте в виду: создание новой ФС уничтожит существующую, поэтому внимательно выбирайте раздел для форматирования.

Комментарий

GParted — одно из лучших приложений с графическим интерфейсом в любой категории. Фактически GParted — это хорошо организованный интерфейс к инструментам командной строки для управления разделами и файловыми системами, который упрощает и ускоряет выполнение сложных задач.

GParted отображает типы файловых систем на смонтированных и размонтированных томах, показывая по одному диску за раз (рис. 9.16). Щелкните на раскрывающемся списке вверху справа, чтобы просмотреть доступные устройства.

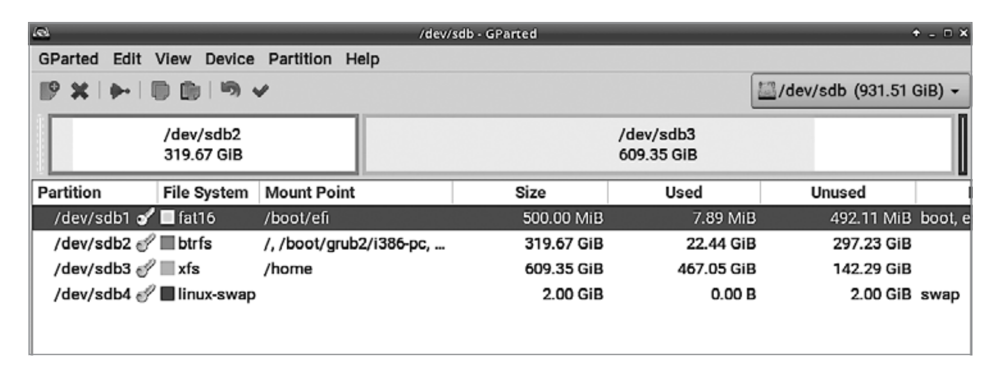


Рис. 9.16. GParted отображает типы файловых систем

ГЛАВА 10

Получение подробной информации об оборудовании компьютера

В Linux есть несколько замечательных утилит для получения подробной информации об аппаратных компонентах компьютера. Вы можете за считаные минуты получить список компонентов компьютера и сведения об их характеристиках, не вскрывая корпус.

С помощью этих утилит можно получить подробную информацию для передачи технической поддержке, поиска правильных драйверов устройств и выяснения, поддерживается ли то или иное устройство в Linux. Конечно, нельзя рассчитывать на то, что производители будут своевременно обновлять информацию о своей продукции. Например, иногда они меняют наборы микросхем, не меняя номера моделей, что может превратить устройство, нормально работавшее в Linux, в устройство, которое там не работает. К счастью, в настоящее время хлопот с поддержкой устройств в Linux стало гораздо меньше, чем раньше.

В идеале у вас также должна иметься документация для вашего компьютера или по крайней мере руководство по материнской плате. Такие руководства обычно изобилуют фотографиями, схемами и полезной информацией и должны быть доступны в Интернете.

В этой главе вы познакомитесь с командами `lshw`, `lspci`, `hwinfo`, `lsusb`, `lscpu` и `lsblk`.

Команды `lshw` и `hwinfo` выводят наиболее полную информацию:

- `lshw` сообщает о конфигурации памяти, версиях прошивки, конфигурации материнской платы, версии и скорости процессора, конфигурации кэша, скорости шины, аппаратных связях, подключенных устройствах, разделах и файловых системах;

- `hwinfo` выводит сведения о мониторе компьютера, RAID-массивах, конфигурации памяти, процессоре, прошивке, конфигурации материнской платы, кэшах, скоростях шины, подключенных устройствах, разделах и файловых системах.

Команда `lsusb` проверяет шины USB и выводит информацию о подключенных к ним устройствах.

`lspci` проверяет шины PCI и выводит информацию о подключенных к ним устройствах.

`lsblk` перечисляет физические диски, разделы и файловые системы.

Команда `lscpu` выводит информацию о процессоре.

10.1. Сбор информации об оборудовании с помощью команды lshw

Задача

Нужно получить перечень оборудования в системе и подробную информацию о каждом компоненте.

Решение

Попробуйте запустить команду `lshw` (hardware lister — «список оборудования») без параметров и переадресовать вывод в текстовый файл:

```
$ sudo lshw | tee hardware.txt
duchess
    description: Laptop
    product: Latitude E7240 (05CA)
    vendor: Dell Inc.
    version: 00
    serial: 456ABC1
    width: 64 bits
[...]
```

Вы получите список в несколько сотен строк, включающий версию прошивки, установленные драйверы, поддерживаемые возможности, серийные номера, номера версий и информацию о шине. Команда `lshw` не проверяет устройства, подключенные через беспроводной интерфейс, например беспроводной принтер или смартфон, подключенный через Bluetooth, но сообщит об имеющихся беспроводных интерфейсах и интерфейсах Bluetooth.

Возможно, вы предпочтете сводку в виде дерева путей к оборудованию:

```
$ sudo lshw -short
H/W path          Device             Class          Description
============================================================
                                     system         To Be Filled By O.E.M.
/0                                   bus            H97M Pro4
/0/0                                 memory         64KiB BIOS
/0/b                                 memory         16GiB System Memory
/0/b/0                               memory         DIMM [empty]
/0/b/1                               memory         8GiB DIMM DDR3 Synchronous
1333
MHz (0.8 ns)
[...]
/0/100/14/0/5                        bus            USB3.0 Hub
/0/100/14/0/5/1                      generic        SAMSUNG_Android
/0/100/14/0/5/2                      printer        MFC-J5945DW
/0/100/14/0/5/4  wlx9cefd5fe8f20    network        802.11 n WLAN
/0/100/14/0/b                        input          USB Optical Mouse
/0/100/14/0/c                        input          QuickFire Rapid keyboard
[...]
```

Или представление по шинам:

```
$ sudo lshw -businfo
Bus info          Device             Class          Description
=============================================================
[...]
cpu@0                                processor      Intel(R) Core(TM) i7-4770K
CPU
@ 3.50GHz
usb@3:5.4         wlx9cefd5fe8f20   network        802.11 n WLAN
usb@3:b                              input          USB Optical Mouse
usb@3:c                              input          QuickFire Rapid keyboard
pci@0000:00:19.0  enp0s25            network        Ethernet Connection (2) I218-V
pci@0000:00:1a.0                     bus            9 Series Chipset Family USB
scsi@0:0.0.0      /dev/sda           disk           4TB ST4000DM000-1F21
scsi@0:0.0.0,1    /dev/sda1          volume         476MiB EXT4 volume
[...]
```

Команда `lshw` имеет также графический интерфейс, который можно открыть с помощью команды `sudo lshw -X`. Часто он устанавливается как отдельный пакет, например *lshw-gtk* в Ubuntu и *lshw-gui* в openSUSE и Fedora.

Комментарий

Команда `lshw` выводит большой объем информации. Посетите главную страницу Hardware Lister (`lshw`) (https://oreil.ly/XRGx1), чтобы узнать о значении разных ее частей.

Команда `lshw` не обнаруживает интерфейсы FireWire или мониторы.

В примере говорится: system To Be Filled By O.E.M. (система заполняется изготовителем), поскольку эта машина была собрана своими руками. Фирменный компьютер, такой как Lenovo или Dell, должен иметь торговую марку и модель.

Столбец `H/W path` содержит аппаратные пути, аналогичные путям в файловой системе. `/0` — это `/system/bus`, что означает компьютер и материнскую плату. Все последующие записи отображаются в виде дерева, похожего на дерево файлов. Как можно видеть в выводе примера, `/0/0` — это `/system/bus/BIOS memory`, `/0/b` — первый заполненный слот ОЗУ, а `/0/b/1` — второй заполненный слот ОЗУ. Эти пути соответствуют физическим соединениям на материнской плате и обычно называются *слотами* или *разъемами*, хотя большинство из них жестко впаяны в материнскую плату и не имеют разъемов, в которые можно было бы вставлять карты расширения.

Дополнительная информация

- Главная страница Hardware Lister (`lshw`) (https://oreil.ly/axiyL).
- `man 1 lshw`

10.2. Фильтрация вывода lshw

Задача

Команда `lshw` выводит действительно большой объем информации, и было бы желательно ограничить его только нужными сведениями.

Решение

Запустите команду `sudo lshw -short` или `sudo lshw -businfo`, чтобы увидеть список классов устройств, а затем передайте команде классы, которые вы хотели бы видеть в выводе:

```
$ sudo lshw -short -class bus -class cpu
```

Опустите параметр `-short`, чтобы получить подробную информацию.

Преобразуйте вывод в формат HTML, XML или JSON и сохраните его в файл, чтобы потом обработать полученные результаты с помощью своих любимых сценариев анализа:

```
$ sudo lshw -html -class bus -class cpu | tee lshw.html
$ sudo lshw -xml -class printer -class display -class input | tee lshw.xml
$ sudo lshw -json -class storage | tee lshw.json
```

Добавьте параметр `-sanitize`, чтобы исключить из вывода конфиденциальную информацию, такую как IP-адреса и серийные номера, чтобы потом ее без опаски можно было передать службе технической поддержки:

```
$ sudo lshw -json -sanitize -class bus -class cpu | tee lshw.json
```

Комментарий

Команда `tee` обеспечивает вывод результатов одновременно на экран и в текстовый файл.

Дополнительная информация

- Главная страница Hardware Lister (`lshw`) (https://oreil.ly/axiyL).
- `man 1 lshw`

10.3. Определение оборудования, включая дисплеи и дисковые массивы RAID, с помощью команды hwinfo

Задача

Нужно получить информацию о дисплее и массивах RAID, а также о других устройствах в системе.

Решение

Команда `hwinfo` сообщает подробные сведения об оборудовании, включая дисплеи и массивы RAID. Следующий пример демонстрирует получение информации о дисплее:

```
$ hwinfo --monitor
[...]
  Hardware Class: monitor
  Model: "VIEWSONIC VX2450 SERIES"
  Vendor: VSC "VIEWSONIC"
  Device: eisa 0xe226 "VX2450 SERIES"
[...]
```

Полный вывод немного длиннее, чем в этом примере, и включает все поддерживаемые разрешения экрана, дату изготовления, диапазоны синхронизации, тип монитора и частоты обновления.

Еще одна важная функция — обнаружение устройств RAID. По умолчанию команда `hwinfo` не пытается обнаруживать их, поэтому используйте параметр `--listmd`:

```
$ hwinfo --listmd
```

Если с этим параметром команда ничего не вывела, значит, в системе нет массивов RAID. В противном случае она выведет большой объем информации.

Ниже показано, как можно получить сводку об имеющемся оборудовании:

```
$ hwinfo --short
keyboard:
  /dev/input/event4    CM Storm QuickFire Rapid keyboard
mouse:
  /dev/input/event5    CM Storm QuickFire Rapid keyboard
  /dev/input/mice      Logitech Optical Wheel Mouse
printer:
                       Brother Industries MFC-J5945DW
monitor:
                       VIEWSONIC VX2450 SERIES
graphics card:
                       Intel Xeon E3-1200 v3/4th Gen Core Processor Integrated
[...]
```

Получить подробную информацию об отдельных компонентах можно следующим образом:

```
$ hwinfo --mouse --network --cdrom
```

Список названий устройств можно увидеть в руководстве `man 8 hwinfo` или выполнив команду `hwinfo --help`:

```
$ hwinfo --help
Usage: hwinfo [OPTIONS]
Probe for hardware.
Options:
    --<HARDWARE_ITEM>
        This option can be given more than once. Probe for a particular
        HARDWARE_ITEM. Available hardware items are:
        all, arch, bios, block, bluetooth, braille, bridge, camera,
        cdrom, chipcard, cpu, disk, dsl, dvb, fingerprint, floppy,
        framebuffer, gfxcard, hub, ide, isapnp, isdn, joystick, keyboard,
        memory, mmc-ctrl, modem, monitor, mouse, netcard, network, partition,
        pci, pcmcia, pcmcia-ctrl, pppoe, printer, redasd,
        reallyall, scanner, scsi, smp, sound, storage-ctrl, sys, tape,
        tv, uml, usb, usb-ctrl, vbe, wlan, xen, zip
[...]
```

Комментарий

Команда `hwinfo` выводит полную и полезную информацию об устройствах. Например, для сетевых интерфейсов она показывает их пути `/sys`, драйверы, состояние канала и MAC-адреса. Для приводов компакт-дисков выводятся название модели, номер версии, драйверы, файлы устройств, скорость привода, список функций и информация о наличии диска в приводе. Команда `hwinfo` часто сообщает больше, чем просто информацию об устройстве как о продукте производителя.

Дополнительная информация

- `man 8 hwinfo`
- Страница команды `hwinfo` на GitHub (https://oreil.ly/BsDAT).

10.4. Определение оборудования PCI с помощью команды lspci

Задача

Получить список устройств, подключенных к шине PCI с названиями производителей и номерами версий.

Решение

Используйте команду `lspci` (list PCI — список устройств PCI). Следующий пример выводит список всех устройств PCI:

```
$ lspci
00:00.0 Host bridge: Intel Corporation 4th Gen Core Processor DRAM Controller
(rev 06)
00:02.0 VGA compatible controller: Intel Corporation Xeon E3-1200 v3/4th Gen
Core Processor Integrated Graphics Controller (rev 06)
00:03.0 Audio device: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor
HD Audio Controller (rev 06)
[...]
```

Чтобы получить больше информации, можно увеличить подробность вывода:

```
$ lspci -v
$ lspci -vv
$ lspci -vvv
```

Увидев сообщение access denied (доступ запрещен), попробуйте выполнить команду `sudo lspci`, чтобы увидеть желаемое.

Комментарий

Команда `lspci` читает информацию с шины PCI, к которой подключены встроенные компоненты на материнской плате, а также карты расширения, вставленные в слоты PCI.

Команда `lspci` отображает дополнительную информацию из собственной базы данных идентификаторов оборудования, например производителей, устройств, классов и подклассов. Эта информация хранится в текстовом файле, но в разных дистрибутивах — в разных местах. В Ubuntu он находится в `/usr/share/misc/pci.ids`, в Fedora — в `/usr/share/hwdata/pci.ids`, а в openSUSE — в `/usr/share/pci.ids`. В руководстве man в вашем дистрибутиве Linux должно быть указано, где находится этот файл, в противном случае попробуйте поискать файл с именем `pci.ids` (`locate pci.ids`).

Разработчики команды `lspci` с радостью примут любую дополнительную информацию; загляните в файл `pci.ids`, где рассказывается, как передать такую информацию, если она у вас есть. Периодически запускайте команду `sudo update-pciids` для обновления базы данных идентификаторов PCI.

Аббревиатура PCI расшифровывается как peripheral component interconnect («соединение периферийных компонентов»). PCI — это локальная аппаратная шина, то есть средство, позволяющее различным аппаратным устройствам компьютера взаимодействовать с ядром Linux. Команда `lspci` в первую очередь пытается обнаружить контроллеры, шины и некоторые отдельные устройства, в том числе:

- контроллеры SATA;
- аудиоконтроллеры и аудиоустройства;
- видеоконтроллеры и видеоустройства;
- контроллеры Ethernet;
- контроллеры USB;
- контроллеры устройств связи;
- контроллеры RAID;
- интегрированные устройства для чтения карт SD/MMC;
- контроллеры PCI FireWire.

За долгие годы сменилось несколько протоколов PCI. Текущий стандарт — PCIe, PCI Express, опубликованный в 2003 году. Он обратно совместим со всеми устаревшими протоколами PCI и заменяет PCI, PCI-X и AGP. Помните AGP — протокол порта ускоренной графики Accelerated Graphics Port? Видеокарты AGP были быстрее видеокарт PCI, так как AGP предоставлял выделенный канал для обработки видео.

PCIe существенно отличается от предшествующих протоколов, поскольку, как и AGP, каждое устройство получает свой отдельный канал. Старые протоколы использовали общую параллельную шину и были значительно медленнее.

Дополнительная информация

- `man 8 lspci`
- `man 8 update-pciids`

10.5. Содержимое вывода команды lspci

Задача

Большая часть вывода команды `lspci` имеет определенный смысл, поскольку это спецификации устройства. Но хотелось бы знать назначение чисел в начале описания каждого устройства, как в данном примере:

```
$ lspci
[...]
00:1f.2 SATA controller: Intel Corporation 9 Series Chipset Family SATA
Controller [AHCI Mode]
[...]
```

Решение

Элемент `00:1f.2` — это номер BDF устройства: (*`шина:устройство.функция`*). Номер шины — 00, номер устройства — 1f и номер функции — 2. Номер функции 2 означает, что устройство выполняет две функции, и каждая из них имеет свой адрес PCI.

Используйте представление в виде дерева, чтобы увидеть, как связаны шина PCI и устройства:

```
$ lspci -tvv
-[0000:00]-+-00.0  Intel Corporation 4th Gen Core Processor DRAM Controller
```

```
+-02.0  Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor
        Integrated Graphics Controller
+-03.0  Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor HD
        Audio Controller
+-14.0  Intel Corporation 9 Series Chipset Family USB xHCI Controller
+-16.0  Intel Corporation 9 Series Chipset Family ME Interface #1
+-19.0  Intel Corporation Ethernet Connection (2) I218-V
+-1a.0  Intel Corporation 9 Series Chipset Family USB EHCI
        Controller #2
+-1b.0  Intel Corporation 9 Series Chipset Family HD Audio Controller
+-1c.0-[01]--
+-1c.3-[02-03]----00.0-[03]--
+-1d.0  Intel Corporation 9 Series Chipset Family USB EHCI
        Controller #1
+-1f.0  Intel Corporation H97 Chipset LPC Controller
+-1f.2  Intel Corporation 9 Series Chipset Family SATA Controller
        [AHCI Mode]
\-1f.3  Intel Corporation 9 Series Chipset Family SMBus Controller
```

Почти во всех персональных компьютерах есть единственная шина PCI, которая всегда имеет номер 00.

Комментарий

Нули в квадратных скобках в корне дерева, [0000: 00], определяют *домен* и *шину*. Первые четыре нуля — это номер домена, а два нуля после двоеточия — номер шины. Домен — это host bridge («главный мост»). Главный мост PCI соединяет контроллер PCI с процессором. «*Домен*» — термин, родившийся в Linux, его наиболее употребительный синоним — «*группа сегментов*». Увидеть его можно, вызвав команду с параметром `-D`:

```
$ lspci -D
0000:00:00.0 Host bridge: Intel Corporation 4th Gen Core Processor DRAM
 Controller (rev 06)
0000:00:02.0 VGA compatible controller: Intel Corporation Xeon E3-1200 v3/4th
Gen
 Core Processor Integrated Graphics Controller (rev 06)
0000:00:03.0 Audio device: Intel Corporation Xeon E3-1200 v3/4th Gen Core
 Processor HD Audio Controller
[...]
```

На серверах с несколькими физическими процессорами можно увидеть несколько главных процессоров, а иногда и несколько шин в одном домене.

Дополнительная информация

- `man 8 lspci`

10.6. Фильтрация вывода команды lspci

Задача

Команда `lspci` выводит действительно большой объем информации, и было бы желательно ограничить его только нужными сведениями.

Решение

Используйте команду `awk` для отсеивания ненужных сведений. Следующий пример оставляет в выводе только записи, относящиеся к USB:

```
$ lspci -v | awk '/USB/,/^$/'
00:14.0 USB controller: Intel Corporation 9 Series Chipset Family USB xHCI
Controller (prog-if 30 [XHCI])
        Subsystem: ASRock Incorporation 9 Series Chipset Family USB xHCI
Controller
        Flags: bus master, medium devsel, latency 0, IRQ 26
        Memory at efc20000 (64-bit, non-prefetchable) [size=64K]
        Capabilities: <access denied>
        Kernel driver in use: xhci_hcd

00:1a.0 USB controller: Intel Corporation 9 Series Chipset Family USB EHCI
Controller #2 (prog-if 20 [EHCI])
        Subsystem: ASRock Incorporation 9 Series Chipset Family USB EHCI
Controller
        Flags: bus master, medium devsel, latency 0, IRQ 16
        Memory at efc3b000 (32-bit, non-prefetchable) [size=1K]
        Capabilities: <access denied>
        Kernel driver in use: ehci-pci
```

Используйте названия классов (Audio, Ethernet, USB и т. д.) в том виде, в каком они появляются в выводе команды `lspci`, и обращайте внимание на регистр, поскольку в команде `awk` довольно трудно организовать поиск без учета регистра. Следующий пример выводит информацию об аудиоконтроллере и устройстве:

```
$ lspci -v | awk '/Audio/,/^$/'
00:03.0 Audio device: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor
HD Audio Controller (rev 06)
        Subsystem: ASRock Incorporation Xeon E3-1200 v3/4th Gen Core Processor
HD Audio Controller
        Flags: bus master, fast devsel, latency 0, IRQ 31
        Memory at efc34000 (64-bit, non-prefetchable) [size=16K]
        Capabilities: <access denied>
        Kernel driver in use: snd_hda_intel
        Kernel modules: snd_hda_intel
```

```
00:1b.0 Audio device: Intel Corporation 9 Series Chipset Family HD Audio
Controller
        Subsystem: ASRock Incorporation 9 Series Chipset Family HD Audio
Controller
        Flags: bus master, fast devsel, latency 0, IRQ 32
        Memory at efc30000 (64-bit, non-prefetchable) [size=16K]
        Capabilities: <access denied>
        Kernel driver in use: snd_hda_intel
        Kernel modules: snd_hda_intel
```

Настройте уровень подробности вывода под свои потребности.

Кроме того, можно выбирать элементы по названию производителя, устройства или номеру класса. Найдите эти числа с помощью параметра `-nn`. В данном примере `0300` (в квадратных скобках) — это номер класса, `8086` — номер производителя, а `0412` — номер устройства:

```
$ lspci -nn
[....]
00:02.0 VGA compatible controller [0300]: Intel Corporation
Xeon E3-1200 v3/4th Gen Core Processor Integrated Graphics Controller
[8086:0412] (rev 06)
[...]
```

Следующие примеры демонстрируют фильтрацию по классу, производителю и устройству соответственно:

```
$ lspci -d ::0604
00:1c.0 PCI bridge: Intel Corporation 9 Series Chipset Family PCI Express Root
Port 1 (rev d0)
00:1c.3 PCI bridge: Intel Corporation 82801 PCI Bridge (rev d0)
02:00.0 PCI bridge: ASMedia Technology Inc. ASM1083/1085 PCIe to PCI Bridge (rev
03)

$ lspci -d 8086::
00:00.0 Host bridge: Intel Corporation 4th Gen Core Processor DRAM Controller
(rev 06)
00:02.0 VGA compatible controller: Intel Corporation Xeon E3-1200 v3/4th Gen
Core Processor Integrated Graphics Controller (rev 06)
00:03.0 Audio device: Intel Corporation Xeon E3-1200 v3/4th Gen Core Processor
HD Audio Controller (rev 06)
[...]

$ lspci -d :0412:
00:02.0 VGA compatible controller: Intel Corporation Xeon E3-1200 v3/4th Gen
Core Processor Integrated Graphics Controller (rev 06)
```

Определить числа для поиска можно, заглянув в репозиторий идентификаторов PCI (https://oreil.ly/f2EKi).

Комментарий

Команда awk — замечательный и очень эффективный инструмент для извлечения определенных текстовых строк из вывода команд или документов. Символ каретки (^) — это якорь регулярного выражения, которому соответствует начало строки, а $ — конец, поэтому регулярному выражению /^$/ в примере выше будут соответствовать разрывы строк, пустые места в начале и конце блока текста. Это известный трюк для извлечения текстовых блоков из источников, в которых имеются пустые строки между разделами.

Дополнительная информация

- man 1 grep
- man 8 lspci
- Репозиторий идентификаторов PCI (https://oreil.ly/f2EKi).

10.7. Использование команды lspci для идентификации модулей ядра

Задача

Узнать, какие модули ядра используют устройства PCI и какие доступны в системе.

Решение

Используйте параметр -k. Следующий пример запрашивает данные для контроллера Ethernet:

```
$ lspci -kd ::0200
00:19.0 Ethernet controller: Intel Corporation Ethernet Connection (2) I218-V
        Subsystem: ASRock Incorporation Ethernet Connection (2) I218-V
        Kernel driver in use: e1000e
        Kernel modules: e1000e
```

Для фильтрации можно использовать команду awk, как в следующем примере, получающем информацию о графическом контроллере:

```
$ lspci -vmmk| awk '/VGA/,/^$/'
Class:  VGA compatible controller
Vendor: Intel Corporation
Device: Xeon E3-1200 v3/4th Gen Core Processor Integrated Graphics Controller
```

```
SVendor:        ASRock Incorporation
SDevice:        Xeon E3-1200 v3/4th Gen Core Processor Integrated Graphics
Controller
Rev:    06
Driver: i915
Module: i915
```

Комментарий

Параметр `-k` показывает используемые, а также доступные модули ядра для каждого устройства. Обычно используемые и доступные модули совпадают, но иногда бывает доступно несколько модулей.

Применяя команду `awk`, не забудьте увеличить уровень подробности вывода, иначе можно не увидеть нужную информацию. См. подраздел «Комментарий» в рецепте 10.6, где приводится дополнительная информация о параметрах этой команды.

Дополнительная информация

- `man 1 awk`
- `man 8 lspci`

10.8. Вывод списка устройств USB с помощью команды lsusb

Задача

Нужен простой и быстрый инструмент для получения списка устройств USB.

Решение

Команда `lsusb` перечисляет шины USB и подключенные к ним устройства, включая мыши, клавиатуры, USB-накопители, принтеры, смартфоны и другие периферийные устройства. В следующих примерах показаны два разных представления одних и тех же устройств.

Запустите команду `lsusb` без параметров, чтобы получить краткую сводку об устройствах USB в вашей системе. Следующий пример был получен в системе с тремя подключенными внешними устройствами USB: клавиатурой, мышью и беспроводным сетевым интерфейсом:

```
$ lsusb
[...]
Bus 003 Device 011: ID 148f:5372 Ralink Technology, Corp. RT5372 Wireless Adapter
Bus 003 Device 002: ID 0bda:5401 Realtek Semiconductor Corp. RTL 8153 USB 3.0
 hub with gigabit ethernet
Bus 003 Device 006: ID 046d:c018 Logitech, Inc. Optical Wheel Mouse
Bus 003 Device 005: ID 2516:0004 Cooler Master Co., Ltd. Storm QuickFire Rapid
 Mechanical Keyboard
[...]
```

Следующий пример сообщает о тех же устройствах, но более подробно и в виде иерархии, включающей драйверы ядра, коды устройств и номера производителей и портов:

```
$ lsusb -tv
[...]
/:  Bus 03.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/14p, 480M
    ID 1d6b:0002 Linux Foundation 2.0 root hub
    |__ Port 3: Dev 2, If 0, Class=Hub, Driver=hub/4p, 480M
        ID 0bda:5401 Realtek Semiconductor Corp. RTL 8153 USB 3.0 hub with
        gigabit ethernet
    |__ Port 7: Dev 11, If 0, Class=Vendor Specific Class, Driver=rt2800usb, 480M
        ID 148f:5372 Ralink Technology, Corp. RT5372 Wireless Adapter
    |__ Port 11: Dev 5, If 0, Class=Human Interface Device, Driver=usbhid, 1.5M
        ID 2516:0004 Cooler Master Co., Ltd. Storm QuickFire Rapid Mechanical
        Keyboard
    |__ Port 12: Dev 6, If 0, Class=Human Interface Device, Driver=usbhid, 1.5M
        ID 046d:c018 Logitech, Inc. Optical Wheel Mouse
[...]
```

В следующих примерах показано, как меняется вывод после подключения внешнего USB-концентратора с интерфейсом Bluetooth и подключенным смартфоном Samsung:

```
$ lsusb
[...]
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 003 Device 012: ID 04e8:6860 Samsung Electronics Co., Ltd Galaxy series,
 misc. (MTP mode)
Bus 003 Device 013: ID 0a12:0001 Cambridge Silicon Radio, Ltd Bluetooth Dongle
 (HCI mode)
Bus 003 Device 002: ID 0bda:5401 Realtek Semiconductor Corp. RTL 8153 USB 3.0
 hub with gigabit ethernet
[...]

$ lsusb -tv
[...]
/:  Bus 03.Port 1: Dev 1, Class=root_hub, Driver=xhci_hcd/14p, 480M
    ID 1d6b:0002 Linux Foundation 2.0 root hub
    |__ Port 3: Dev 2, If 0, Class=Hub, Driver=hub/4p, 480M
```

```
        ID 0bda:5401 Realtek Semiconductor Corp. RTL 8153 USB 3.0 hub with
        gigabit ethernet
        |__ Port 4: Dev 12, If 0, Class=Imaging, Driver=, 480M
            ID 04e8:6860 Samsung Electronics Co., Ltd Galaxy series, misc. (MTP
            mode)
        |__ Port 2: Dev 13, If 0, Class=Wireless, Driver=btusb, 12M
            ID 0a12:0001 Cambridge Silicon Radio, Ltd Bluetooth Dongle (HCI mode)
        |__ Port 2: Dev 13, If 1, Class=Wireless, Driver=btusb, 12M
            ID 0a12:0001 Cambridge Silicon Radio, Ltd Bluetooth Dongle (HCI mode)
[...]
```

Комментарий

Номера шины и порта всегда совпадают. Номер устройства меняется при каждом подключении устройства.

Идентификационные номера, например `0a12:0001`, состоят из кода производителя и кода устройства. Производители должны обращаться на https://usb.org для получения новых кодов. Список текущих USB-идентификаторов можно найти на linux-usb.org (https://oreil.ly/bHLo6) и там же можно оставить обновленную информацию.

Коды классов также управляются https://usb.org; см. коды классов USB (https://oreil.ly/vNCgT). Мне показалось интересным, что Dev 57 — телефон Samsung на базе Android — классифицируется как устройство для обработки изображений. Впрочем, такая классификация не лишена смысла, поскольку большинство дистрибутивов Linux используют протокол передачи мультимедиа (MTP) для обмена файлами с телефонами на Android.

Примеры в этом разделе получены на персональном компьютере, на котором имеются порты USB 2.0 и USB 3.1. Вывод команды `lsusb` показывает скорости, согласованные с устройствами, так что когда вы видите нечто наподобие `usbhid, 1.5M` вместо `480M` или `5000M` — это нормально, поскольку это клавиатура, которой не нужна полная скорость USB-соединения. Для устройств хранения, таких как USB-накопители и внешние жесткие диски, вы почти наверняка увидите более высокие скорости.

Дополнительная информация

- `man 8 lsusb`
- USB-IF (https://usb.org).
- Список USB ID (https://oreil.ly/js1oj).

10.9. Вывод списка разделов и жестких дисков с помощью команды lsblk

Задача

Нужен простой и быстрый инструмент для получения списка всех подключенных устройств хранения и их разделов.

Решение

Используйте команду `lsblk` (list block devices — «список блочных устройств»). Запустите ее без параметров, чтобы получить список всех блочных устройств на компьютере:

```
$ lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sda      8:0    0   3.7T  0 disk
├─sda1   8:1    0   476M  0 part /boot
├─sda2   8:2    0  55.9G  0 part /
├─sda3   8:3    0   1.8T  0 part /home
└─sda4   8:4    0   7.5G  0 part [SWAP]
sdb      8:16   0   1.8T  0 disk
├─sdb1   8:17   0   102M  0 part
├─sdb2   8:18   0   6.5G  0 part
├─sdb3   8:19   0   1.1G  0 part [SWAP]
└─sdb4   8:20   0   1.8T  0 part
sdc      8:32   0   3.7T  0 disk
├─sdc1   8:33   0   128M  0 part
├─sdc2   8:34   0 439.7G  0 part
└─sdc3   8:35   0   3.2T  0 part
sdd      8:48   1   3.8G  0 disk
└─sdd1   8:49   1   3.8G  0 part
sr0     11:0    1 159.3M  0 rom
```

Ниже показано, как можно вывести метки файловых систем и идентификаторы UUID выбранного устройства:

```
$ lsblk -f /dev/sdc
NAME   FSTYPE LABEL               UUID               MOUNTPOINT
sdc
├─sdc1
├─sdc2 ntfs   Seagate Backup Plus  2E203F82203F5057
└─sdc3 ext4   backup               0451d428-9716-4cdd  /media/max/backup
```

Список только SCSI-устройств и их типы:

```
$ lsblk -S
NAME HCTL       TYPE VENDOR   MODEL             REV TRAN
sda  0:0:0:0    disk ATA      ST4000DM000-1F21 CC54 sata
sdb  2:0:0:0    disk ATA      SAMSUNG HD204UI  0001 sata
sdc  6:0:0:0    disk Seagate  BUP SL           0304 usb
sr0  4:0:0:0    rom  ATAPI    iHAS424   B      GL1B sata
```

Комментарий

Сочетания `sda` и `sdb` обозначают жесткие диски SATA, а `sdc` — USB-накопитель. В Linux устройства хранения данных, такие как жесткие диски SATA и флеш-накопители, обслуживаются драйвером SCSI. Сочетания `sr0`, `rom` и `ATAPI` идентифицируют проигрыватель CD/DVD.

Дать определение термина *«блочные устройства»* без начального отступления довольно сложно, поскольку это программный термин, который трудно объяснить кратко, аппелируя к привычным пользователю понятиям. По своему опыту могу сказать, что полезнее рассматривать блочные устройства как запоминающие устройства большой емкости и как разделы на этих устройствах.

- `MAJ:MIN` обозначает старший и младший номера. Старший определяет категорию, например, 8 соответствует устройствам `sd`, а младший обозначает каждое устройство в последовательности. (Запустите **`lsblk -l`**, чтобы увидеть древовидную иерархию устройств.)
- `RM` сообщает, является ли это устройство съемным, значение `1` в данном столбце соответствует съемным устройствам.
- `SIZE` — размер блочного устройства.
- `RO = 0` означает, что устройство доступно не только для чтения, а `1` — только для чтения. Привод CD/DVD sr0 доступен для чтения и записи, но команда `lsblk` не может сказать, доступен ли для записи сам диск в устройстве sr0.
- `TYPE` определяет тип диска.
- `MOUNTPOINT` показывает пути к смонтированным устройствам.

Дополнительная информация

- `man 8 lsblk`

10.10. Получение информации о процессоре

Задача

Узнать, какой процессор или процессоры имеются в системе, и получить информацию о них.

Решение

Выполните команду `lscpu` (list CPU — «список процессоров») без параметров:

```
$ lscpu
Architecture:        x86_64
CPU op-mode(s):      32-bit, 64-bit
Byte Order:          Little Endian
CPU(s):              8
On-line CPU(s) list: 0-7
Thread(s) per core:  2
Core(s) per socket:  4
Socket(s):           1
Vendor ID:           GenuineIntel
CPU family:          6
Model:               60
Model name:          Intel(R) Core(TM) i7-4770K CPU @ 3.50GHz
[...]
L1d cache:           128 KiB
L1i cache:           128 KiB
L2 cache:            1 MiB
L3 cache:            8 MiB
[...]
```

Она выведет большой объем информации; вы также увидите большое количество флагов поддерживаемых возможностей и сведения о встроенных кэшах.

Комментарий

Современные процессоры оснащаются кэшами трех типов: L1, L2 и L3. Это небольшие модули кэш-памяти внутри процессора. Они очень быстрые, во много раз быстрее системной оперативной памяти, и хранят данные, которые, скорее всего, понадобятся процессору при выполнении последующих операций. L1 — самый быстрый и дорогой кэш, поэтому обычно он самый маленький. L2 — следующий по быстродействию и менее дорогой кэш, обычно большего объема, чем L1. Кэш L3 — самый медленный и наименее дорогостоящий, обычно самого большого объема.

Как показано в предыдущем примере, процессор имеет четыре кэша. Используйте параметр `-C`, чтобы получить больше информации о кэше:

```
$ lscpu -C
NAME ONE-SIZE ALL-SIZE WAYS TYPE        LEVEL
L1d       32K     128K    8 Data            1
L1i       32K     128K    8 Instruction     1
L2       256K       1M    8 Unified         2
L3         8M       8M   16 Unified         3
```

Здесь можно видеть четыре кэша, используемых четырьмя физическими ядрами процессора. Кэш L1i предназначен для кэширования инструкций процессора, а L1d — данных. L2 и L3 хранят только данные.

Количество ядер процессора может немного сбивать с толку. Строка `CPU(s): 8` в этом примере не означает, что данный процессор имеет восемь физических ядер, просто такое количество ядер «видит» ядро Linux. Следующие строки поясняют суть:

```
Thread(s) per core: 2
Core(s) per socket: 4
Socket(s): 1
```

Как видите, это единственный процессор с четырьмя физическими ядрами и двумя потоками на ядро, всего получается восемь логических процессоров.

Дополнительная информация

- `man 1 lscpu`

10.11. Идентификация аппаратной архитектуры

Задача

Вы не уверены в том, какова аппаратная архитектура вашей машины; вы думаете, что это x86-64 или ARM, но хотели бы знать точно.

Решение

Используйте команду `uname`. Ниже представлен пример, полученный на машине x86-64:

```
$ uname -m
x86_64
```

Вот список наиболее распространенных результатов, которые можно увидеть:

- arm;
- aarch64;

- armv7* (arm7 и ниже — это 32-битные архитектуры);
- armv8* (arm8 и выше — 64-битные архитектуры);
- ia64;
- ppc;
- ppc64;
- s390x;
- sparc;
- sparc64;
- i386;
- i686;
- x86_64.

Если машина работает не под управлением Linux, то попробуйте загрузиться с USB-накопителя SystemRescue, а затем выполните команду `uname -m`.

Вы можете установить Linux на Chromebook. В Chromebook используются процессоры Intel и ARM. Один из способов определить тип процессора — открыть в браузере страницу chrome://system. На ней вы увидите всю системную информацию, возможно, даже больше, чем хотелось бы.

Более удобный инструмент — Cog System Info Viewer (https://oreil.ly/Yeirk). Он отображает информацию об оборудовании и сети на Chromebook.

Комментарий

Linux поддерживает больше аппаратных архитектур, чем любая другая ОС, от крошечных встраиваемых систем и систем на кристалле (System on a Chip, SoC) до мейнфреймов и суперкомпьютеров и всего, что между ними. Какое бы непонятное компьютерное оборудование вы бы ни использовали, есть вероятность, что на нем будет работать та или иная разновидность Linux.

Дополнительная информация

- `man 1 uname`

ГЛАВА 11

Создание файловых систем и управление ими

Количество файловых систем, поддерживаемых в Linux, больше, чем в любой другой операционной системе. ФС необходимы для вычислений и выполняют поразительный объем работы. Файловая система компьютера хранит, упорядочивает и защищает наши данные и все время находится под нагрузкой из-за постоянного применения. Нам, пользователям Linux, повезло, что мы можем выбирать из множества замечательных файловых систем.

В этой главе вы познакомитесь с инструментами командной строки для создания и управления следующими файловыми системами общего назначения, которые полностью поддерживаются в Linux:

- Ext4, Extended Filesystem (расширенная файловая система);
- XFS, X File System (файловая система X); буква X ничего не обозначает, это просто X;
- Btrfs, b-tree filesystem (файловая система на основе бинарных деревьев), произносится как «баттер эф-эс»;
- FAT16/32, File Allocation Table (таблица распределения файлов), 16- и 32-битная;
- exFAT, Extended FAT (расширенная FAT), новейшая 64-битная файловая система компании Microsoft.

В эту главу не включены файловые системы NTFS от Microsoft и HFS/HFS+/APFS от Apple. Linux хорошо поддерживает NTFS как для чтения, так и для записи. Чтобы попробовать, поищите пакеты *ntfs-3g* (NTFS третьего поколения).

Поддержка файловой системы HFS/HFS+/APFS компании Apple пока ненадежна. Чтобы протестировать ее, найдите пакеты с сочетаниями hfs или apfs в именах и убедитесь, что в описании к ним указано, что они реализуют поддержку файловых систем Apple.

Существует множество специализированных ФС, таких как UBIFS и JFFS2 для устройств CompactFlash; сжатая файловая система SquashFS, HDFS, CephFS и GlusterFS для распределенных вычислений; NFS для обмена файлами по сети и многие другие. Их описание легко может занять отдельную большую книгу, и поэтому они не были включены сюда. Эти файловые системы находятся в свободном доступе, и вы можете попробовать их и изучить самостоятельно.

Обзор файловых систем

Прежде чем использовать какое-либо устройство хранения, например жесткий диск, USB-накопитель или SD-карту, его нужно разбить на разделы и отформатировать в файловую систему. Каждая файловая система должна иметь свой раздел на диске. Раздел может охватывать весь диск, или же диск может быть разделен на несколько разделов. Каждый раздел подобен независимому диску и может иметь свою файловую систему.

ФС должна быть смонтирована (подключена) в работающую файловую систему, и только потом она станет доступной. Файловой системе нужна *точка монтирования* — каталог, созданный специально для монтирования этой системы. Такие каталоги могут находиться где угодно, хотя обычно используются `/mnt` и `/media`.

В каждую точку монтирования можно смонтировать только одну файловую систему. Если в ту же точку смонтировать вторую ФС, она сделает недоступной первую.

Файловые системы могут монтироваться автоматически — при запуске системы, динамически — при подключении съемного носителя, вручную — из командной строки или нажатием кнопки на рабочем столе или в диспетчере файлов. Большинство дистрибутивов Linux заботятся об удобстве работы со съемными носителями. Подключите USB-устройство или вставьте оптический диск, и операционная система Linux сама настроит необходимые точки монтирования и смонтирует их; также можно настроить систему для монтирования файловых систем одним щелчком кнопкой мыши (рис. 11.1).



Рис. 11.1. Кнопки монтирования съемных устройств на рабочем столе Xfce

Ext4, XFS, Btrfs и exFAT — это 64-битные файловые системы. То есть они поддерживают 64-битное пространство адресации блоков, что позволяет иметь файлы и файловые системы гораздо большего размера, чем 32- и 16-битные файловые системы. Шестидесятичетырехбитные вычисления используются как минимум с 1970-х годов. Сначала они были реализованы на суперкомпьютерах, а затем и на высокопроизводительных коммерческих ЭВМ, таких как IBM Power и Sun Microsystems UltraSPARC.

Мой первый компьютер с Windows 3.1/DOS еще в середине 1990-х был 16-битной системой. Windows 95 хвасталась тем, что стала первой 32-битной потребительской операционной системой. Первые 64-битные файловые системы для компьютеров x86 начали появляться в Linux примерно в 2001 году. В главе High Level Design в документации ядра Linux с описанием Ext4 (https://oreil.ly/kufyJ) приводятся наглядные таблицы сравнения 32- и 64-битных файловых систем.

Шестидесятичетырехбитные файловые системы обратно совместимы с 32-битными приложениями. По прошествии стольких лет маловероятно, что вы столкнетесь с 32-битными приложениями, однако если это случится, то они будут работать в вашем современном Linux при условии, что в нем установлены все необходимые пакеты для поддержки 32-битной среды.

Ext4 и XFS — журналируемые файловые системы, а Btrfs — файловая система с *копированием при записи* (copy-on-write, CoW). Журналирование и CoW обеспечивают сохранность файловых систем в согласованном состоянии даже после сбоя питания или краха системы. ФС сложны и постоянно нагружены,

поэтому любое неожиданное прерывание их работы влияет не только на файлы, с которыми вы работаете. Сбои приводят к появлению большого количества файлов, операции с которыми остались незавершенными, а в былые времена это приводило даже к потере всей файловой системы.

Ext4 — наиболее широко используемая файловая система в Linux и предлагается по умолчанию в большинстве дистрибутивов. Она хорошо протестирована, имеет хорошую поддержку и выполняет свою работу без драматизма. Журнал Ext4 записывает изменения до их сохранения на диск, обеспечивая защиту от потери данных в случае прерывания работы. Размер файловой системы Ext4 можно изменять как в большую, так и в меньшую сторону.

XFS первоначально была одной из самых высокопроизводительных 64-битных файловых систем в Unix и перенесена в Linux в 2001 году. Это быстрая, эффективная и надежная журналируемая ФС, подходящая для самых разных систем — от небольших персональных компьютеров до многодисковых центров обработки данных. Размер XFS можно увеличивать, но нельзя уменьшать.

Btrfs — это продвинутая файловая система с механизмом копирования при записи (copy-on-write, CoW), включающим набор функций, которые отсутствуют в других ФС, описываемых в этой главе, таких как создание моментальных снимков, поддержка RAID 0, 1 и 10, а также и подтома (subvolumes). Подтома — удивительно гибкая штука, они позволяют создать несколько корней в файловой системе в одном разделе. Копирование при записи (CoW) — отличный способ создавать моментальные снимки, экономно используя место, когда каждый снимок содержит только изменения относительно предыдущего снимка. Если у вас возникнут проблемы, то вы сможете вернуться к более старому, заведомо исправному снимку. Размер Btrfs можно изменять как в большую, так и в меньшую сторону.

FAT16/32 — устаревшие 16- и 32-разрядные файловые системы компании Microsoft. FAT32 — наиболее универсальная ФС, поддерживаемая операционными системами Microsoft Windows, Apple MacOS, Linux, Unix и DOS. Используйте FAT32 на съемных носителях, если вам нужно облегчить обмен файлами. Но у нее есть одно ограничение, являющееся препятствием для некоторых вариантов применения, а именно — максимальный размер файла 4 Гбайт (на носителях с блоками по 4 Кбайт).

exFAT — новейшая 64-битная файловая система Microsoft, хорошее обновление устаревшей FAT32. exFAT — быстрая и легкая ФС, прекрасно подходит для USB-накопителей и SD-карт и поддерживает файлы и тома гораздо большего

размера, чем FAT32. «Википедия» сообщает, что максимальный размер файла в этой файловой системе составляет 16 ЭиБ, а максимальный размер тома — 128 ПиБ. У нее нет журнала или механизма CoW.

exFAT мало распространена среди пользователей Linux, поскольку это запатентованная проприетарная ФС, которая была недоступна для Linux в роли собственной файловой системы до 2020 года. Беспокоиться о совместимости exFAT с Linux приходится только в случае необходимости читать и копировать USB-накопители или карты SDXC, отформатированные в exFAT. Такая необходимость может возникнуть, например, если вам понадобится использовать карты SDXC в формате exFAT с цифровой камерой или с устройством записи звука.

Есть два варианта использования exFAT в Linux. Один из них — установить пакеты *exfatprogs* или *exfatfuse* в паре с *exfat-utils*, которые доступны в большинстве дистрибутивов. exFAT FUSE была разработана и поддерживается за пределами США, поэтому на нее не распространяется патентное право США. exFAT FUSE применяет файловую систему в пользовательском пространстве (Filesystem in Userspace, FUSE), что позволяет запускать ее непривилегированным пользователям. ФС в пользовательском пространстве действует не так эффективно, как должным образом интегрированная в ядро, но работает и позволяет читать и записывать файлы exFAT. Некоторые особенно упорные пытаются использовать exFAT FUSE в общих разделах, чтобы обеспечить возможность обмена файлами с пользователями Windows и macOS. Теоретически это вполне работоспособный подход, хотя иногда возникают сбои, связанные с тем, насколько хорошо конкретная версия Windows или macOS реализует exFAT.

Другой вариант — немного подождать, пока появится встроенная поддержка. Microsoft выпустила exFAT в 2006 году и лицензировала ее в первую очередь для компаний, производящих встраиваемые системы и носители. Но времена меняются. Microsoft тоже начала вносить свой вклад в распространение ПО с открытым исходным кодом и вступила в организацию Open Invention Network (OIN) (https://oreil.ly/AJepb). Microsoft выпустила спецификацию exFAT в 2019 году. Выпуск спецификации позволил избежать проблем с лицензированием существующего кода exFAT, и благодаря этому разработчикам ядра Linux не пришлось терять время на написание нового кода. Новенькая, блестящая встроенная поддержка exFAT появилась в ядре Linux 5.7. Скоро оно должно попасть в ваш любимый дистрибутив; выполните команду `uname -r`, чтобы увидеть версию вашего ядра.

11.1. Вывод списка поддерживаемых файловых систем

Задача

Узнать, поддержка каких файловых систем предусмотрена в системе Linux.

Решение

Прочитайте содержимое файла `/proc/filesystems`:

```
$ cat /proc/filesystems
nodev   sysfs
nodev   bdev
nodev   proc
nodev   cgroup
nodev   cgroup2
nodev   cpuset
nodev   devtmpfs
nodev   debugfs
nodev   tracefs
nodev   securityfs
nodev   sockfs
nodev   bpf
nodev   pipefs
nodev   ramfs
nodev   hugetlbfs
nodev   devpts
        ext3
        ext2
        ext4
nodev   autofs
nodev   mqueue
nodev   pstore
        btrfs
        vfat
        xfs
        fuseblk
nodev   fuse
nodev   fusectl
        jfs
        nilfs2
```

Комментарий

Видите все эти записи `nodev`? Это виртуальные файловые системы, которые существуют только в памяти и не подключены к физическим устройствам, таким как `/dev/sda1`. Всеми этими виртуальными ФС управляет система.

Другие файловые системы, Ext4, XFS и т. д., являются ФС, которые мы используем на наших дисковых устройствах для хранения, организации и защиты данных.

Дополнительная информация

sysfs, the filesystem for exporting kernel objects (https://oreil.ly/QCMN7); статья написана для разработчиков, но содержит информацию, важную для пользователей и администраторов Linux.

11.2. Идентификация существующих файловых систем

Задача

Вы не знаете, какие файловые системы имеются на ваших устройствах хранения или на съемных носителях, и вам нужен способ узнать это.

Решение

Используйте команду `lsblk`. С ее помощью можно получить только имена устройств и названия файловых систем, передав параметры `NAME` и `FSTYPE`:

```
$ lsblk -o NAME,FSTYPE
NAME   FSTYPE
sda
├─sda1 vfat
├─sda2 btrfs
├─sda3 xfs
└─sda4 swap
sdb
├─sdb1 ext2
├─sdb2 ext4
├─sdb3 swap
└─sdb4 LVM2_member
sdc
└─sdc1 vfat
sr0
```

Запросить информацию о конкретном диске можно так:

```
$ lsblk -o NAME,FSTYPE /dev/sdb
├─sdb1 ext2
├─sdb2 ext4
├─sdb3 swap
└─sdb4 LVM2_member
```

или о конкретном разделе:

```
$ lsblk -o NAME,FSTYPE /dev/sda1
NAME FSTYPE
sda1 vfat
```

А вот мое любимое заклинание с командой `lsblk`. Оно показывает все имена устройств, типы и размеры файловых систем, процент заполнения, метки и точки монтирования:

```
$ lsblk -o NAME,FSTYPE,LABEL,FSSIZE,FSUSE%,MOUNTPOINT
NAME   FSTYPE   LABEL     FSSIZE FSUSE% MOUNTPOINT
loop0  squashfs           646.5M   100% /run/archiso/sfs/airootfs
sda
├─sda1
└─sda2 ntfs
sdb
├─sdb1 vfat     BOOT
├─sdb2 btrfs    root
├─sdb3 xfs      home
└─sdb4 swap
sdc    iso9660  RESCUE800
└─sdc1 iso9660  RESCUE800   708M   100% /run/archiso/bootmnt
sr0
```

Комментарий

Выполните команду `lsblk --help`, чтобы увидеть список поддерживаемых столбцов. Их довольно много, в том числе: `PATH`, `LABEL`, `UUID`, `HOTPLUG`, `MODEL`, `SERIAL` и `SIZE`.

В некоторых дистрибутивах могут потребоваться права root для просмотра типов файловых систем, UUID и меток.

Команда `lsblk` всегда печатает *vfat* для файловых систем FAT16 и FAT32. Используйте GParted или `parted`, чтобы узнать точный тип файловой системы — FAT16 или FAT32.

vfat — это Virtual FAT, драйвер файловой системы ядра для FAT16 и FAT32.

Дополнительная информация

- Руководство по интерфейсам SCSI в ядре Linux (https://oreil.ly/beFOx).
- Список старших и младших номеров блочных и символьных устройств (https://oreil.ly/NW2S7).

- `man 8 lsblk`
- `man 8 parted`
- Глава 8.
- Глава 9.

11.3. Изменение размера файловой системы

Задача

Увеличить или уменьшить размер файловой системы.

Решение

Каждая файловая система имеет свои команды для изменения размера. Обращайтесь к рецептам 8.8, 8.9 и 9.7, чтобы узнать больше об изменении размеров ФС.

Комментарий

Размер раздела с файловой системой тоже должен быть изменен соответствующим образом. GParted объединяет эти шаги в одну операцию (см. рецепт 9.7).

В рецептах 8.8 и 8.9 используются `parted` и утилиты для изменения размера файловой системы и ее раздела в два этапа.

Дополнительная информация

- Рецепт 8.8.
- Рецепт 8.9.
- Рецепт 9.7.
- `man 8 resize2fs`
- `man 8 parted`
- `man 8 xfs_growfs`
- `man 8 btrfs`
- `man 8 fsck.vfat`

11.4. Удаление файловых систем

Задача

Удалить файловую систему вместе с разделом.

Решение

Чтобы удалить ФС вместе с разделом, используйте команду `parted`. Ниже приводится пример удаления раздела `/dev/sdb1`. Перепроверьте, какой именно раздел и файловую систему вы собираетесь удалить, затем размонтируйте ФС. В этом примере файловая система смонтирована в каталог `/media/duchess/stuff`:

```
$ lsblk -f
sda
├─sdb1 ext4 /media/duchess/stuff
[...]
$ umount /media/duchess/stuff
```

После этого используйте команду `parted` для удаления раздела:

```
$ sudo parted /dev/sdb
GNU Parted 3.2
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print

Model: ATA SAMSUNG HD204UI (scsi)
Disk /dev/sdb: 2000GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  1656GB  1656GB  ext4          stor-1
 1      1656GB  2656GB  1000GB  ext4          stor-2
(parted) rm 1
```

Если вы предпочитаете инструменты с графическим интерфейсом, то используйте GParted (см. главу 9).

Комментарий

Да, в примере нет опечатки, команда действительно называется `umount`, а не `unmount`. Название `umount` восходит к древней эпохе Unix, когда длина имен файлов была ограничена шестью символами.

Удаление всех файлов не приводит к удалению файловой системы. Структура ФС остается на месте.

Дополнительная информация

- `man 1 dd`

11.5. Использование новой файловой системы

Задача

Вы только что создали новую файловую систему и теперь хотите ее смонтировать.

Решение

После создания новой файловой системы нужно создать точку монтирования и, если необходимо, настроить автоматическое монтирование. Как обсуждалось во введении к этой главе, новая ФС должна быть смонтирована (подключена) в работающую файловую систему, чтобы ее можно было использовать.

Ext4, XFS и Btrfs имеют средства управления доступом. Если нужно, чтобы файлы в этих файловых системах были доступны всем, а не только пользователю root, то необходимо настроить атрибуты владения и разрешения. FAT16/32 и exFAT не имеют средств управления доступом и открыты для всех.

Для начала смонтируйте новую файловую систему. Создайте точку монтирования — каталог, а затем смонтируйте ФС, как показано в следующем примере:

```
$ sudo mkdir -p /mnt/madmax/newfs
$ sudo mount /dev/sdb1 /mnt/madmax/newfs
```

Следующий пример устанавливает пользователя `madmax` владельцем новой файловой системы, выдает разрешения для чтения/записи/выполнения владельцу и разрешения только для чтения группе и всем остальным:

```
$ sudo chown -R madmax:madmax /mnt/madmax/newfs
$ sudo chmod -R 0755 /mnt/madmax/newfs
```

Теперь Безумный Макс (`madmax`) сможет использовать новую файловую систему. В данном случае она останется смонтированной только до перезапуска системы; см. рецепт 11.6, чтобы узнать, как настроить автоматическое монтирование ФС.



Одна точка монтирования — одна файловая система

Каждой файловой системе нужна своя уникальная точка монтирования; не следует монтировать несколько ФС в одну точку монтирования.

Комментарий

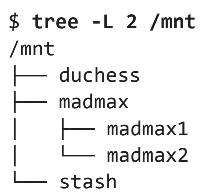
Подробные рецепты управления атрибутами владения и разрешениями вы найдете в главе 6.

Традиционно точки монтирования помещаются в каталоги `/mnt` и `/media`. Каталог `/mnt` используется для статического монтирования (настраивается в `/etc/fstab`), а `/media` — для автоматического монтирования съемных носителей. Вы можете создавать свои точки монтирования где угодно. Преимущество использования традиционных каталогов в том, что точки монтирования находятся в ограниченном количестве предсказуемых мест.

Общий каталог с точками монтирования для нескольких пользователей может выглядеть так, как показано ниже. Здесь каждому пользователю отведен свой каталог для точек монтирования:

```
$ tree /shared
/shared
├── duchess
├── madmax
└── stash
```

Для каждой ФС в подкаталогах пользователей нужно создать свою точку монтирования. Например, Безумный Макс имеет две точки монтирования файловых систем — `madmax1` и `madmax2`:



```
$ tree -L 2 /mnt
/mnt
├── duchess
├── madmax
│   ├── madmax1
│   └── madmax2
└── stash
```

Точки монтирования могут иметь любые имена. Например, точки монтирования Безумного Макса могут иметь имена `fs1` и `fs2`, или `fred` и `ethel`, или `max1` и `max2`, но лучше выбирать такие имена, которые будут напоминать назначение файловой системы.

Увидеть разрешения в файловой системе можно с помощью команды `stat`, как показано в этом примере:

```
$ stat /shared/madmax/madmax1
[...]
Access: (0755/drwxr-xr-x) Uid: ( 0/ madmax) Gid: ( 0/ madmax)
```

Список всех смонтированных файловых систем можно получить с помощью команды `mount`:

```
$ mount
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs
[...]
```

Узнать, является ли каталог точкой монтирования, можно с помощью команды `mountpoint`:

```
$ mountpoint madmax1/
madmax1/ is a mountpoint
```

Дополнительная информация

- `man 1 chown`
- `man 1 chmod`
- `man 1 stat`

11.6. Автоматическое монтирование файловой системы

Задача

Вы добавили новую файловую систему и теперь хотите, чтобы она монтировалась автоматически при запуске системы.

Решение

Для этой цели существует файл `/etc/fstab`. В следующем примере показаны строки, которые нужно добавить в существующий `/etc/fstab`, чтобы обеспечить автоматическое монтирование файловой системы из рецепта 11.5 при запуске системы:

```
#<файловая система> <точка монтирования> <тип>    <параметры>     <дамп>
<порядок проверки>
LABEL=xfs-ehd       /mnt/madmax/newfs    xfs      defaults,user   0       2
```

Используйте команду `findmnt` для проверки новой конфигурации:

```
$ sudo findmnt --verbose --verify
/
   [ ] target exists
   [ ] UUID=102a6fce-8985-4896-a5f9-e5980cb21fdb translated to /dev/sda2
   [ ] source /dev/sda2 exists
   [ ] FS type is btrfs
   [W] recommended root FS passno is 1 (current is 0)
/mnt/madmax/newfs
   [ ] target exists
   [ ] LABEL=xfs-ehd translated to /dev/sdb1
   [ ] source /dev/sdb1 exists
   [ ] FS type is xfs
[...]
0 parse errors, 0 errors, 1 warning
```

Предупреждение recommended root FS passno is 1 (current is 0) (рекомендуемый порядок проверки корневой файловой системы — 1 (текущий — 0) в данном случае можно игнорировать. Если это единственное предупреждение и ошибок нет, то перезагрузитесь или выполните следующую команду, чтобы смонтировать файловую систему, настроенную в новой записи файла `/etc/fstab`:

```
$ sudo mount -a
```

Комментарий

Ниже представлено значение шести полей в записях в файле `fstab`:

- *устройство* — UUID или метка файловой системы. Не используйте имена `/dev`, поскольку они неуникальны и иногда могут изменяться. Выполните команду `lsblk -o UUID, LABEL`, чтобы получить список UUID и меток файловой системы, которые можно использовать в столбце *«устройство»*;
- *точка монтирования* — каталог, созданный вами для монтирования файловой системы;
- *тип* — тип файловой системы, например xfs, ext4 или btrfs. Можно использовать и тип auto, чтобы ядро автоматически определяло тип файловой системы;
- *параметры* — список параметров монтирования через запятую (поддерживаемые параметры перечислены ниже);
- *дамп* — если вы используете команду `dump` для резервного копирования, то число в этом столбце подскажет команде `dump` интервал резервного копирования в днях. Число 1 означает каждый день, 2 — каждый второй день,

3 — каждый третий день и т. д. Скорее всего, вы не используете dump, поэтому укажите в этом поле 0;

- *порядок проверки* — данное число сообщает программе проверки, какую файловую систему проверять в первую очередь при загрузке, если это понадобится. Для корневой файловой системы в этом поле должно быть число 1, для других файловых систем Linux — число 2, а для файловых систем, неродных для Linux, — число 0.

Следующие *параметры* определяют разрешения:

- defaults — включает параметры: rw, suid, dev, exec, auto, nouser и async. Значения, определяемые параметром defaults, можно переопределить добавлением дополнительных параметров, например, defaults,user разрешает монтировать и размонтировать файловую систему простым пользователям. Вы можете добавить сколько угодно параметров или опустить defaults и указать только те параметры, которые вам нужны;
- rw — для чтения записи;
- ro — только для чтения;
- suid — включить поддержку битов setuid и setgid;
- dev — интерпретировать блочные и символьные устройства;
- exec — разрешить запуск двоичных файлов;
- auto — указывает, какие файловые системы должны монтироваться при загрузке;
- nouser — рядовые пользователи не могут монтировать или размонтировать файловую систему;
- async — асинхронный ввод/вывод, стандартный параметр для Linux;
- user — рядовой пользователь может монтировать файловую систему и размонтировать ее, если она была смонтирована им;
- users — любой пользователь может монтировать и размонтировать файловую систему;
- noauto — не монтировать автоматически при загрузке;
- noatime — не обновляйте атрибут файлов «время последнего доступа». В прошлом параметр noatime использовался для повышения производительности. Если у вас современный компьютер, то, вероятно, этот параметр не будет иметь большого значения;
- gid — ограничить доступ группой (из /etc/group); например, gid=group1.

Дополнительная информация

- `man 8 mount`
- `man 5 fstab`
- Systemd (https://systemd.io).

11.7. Создание файловой системы Ext4

Задача

Создать новую файловую систему Ext4 на внутреннем или внешнем диске.

Решение

Сначала создайте раздел соответствующего размера для файловой системы. Затем используйте команду `mkfs.ext4` для создания новой файловой системы Ext4.

В следующем примере существующая файловая система XFS затирается новой файловой системой Ext4. Прежде чем затирать существующую ФС, ее сначала нужно размонтировать. В этом примере файловая система находится на устройстве `/dev/sdb1` и смонтирована в `/media/duchess/stuff`, как показывает команда `df`:

```
$ df -Th /media/duchess/stuff/
Filesystem     Type  Size  Used Avail Use% Mounted on
/dev/sdb1      xfs   952M  7.9M  944M   1% /media/duchess/stuff
```

Чтобы размонтировать файловую систему, нужны привилегии root:

```
$ sudo umount /media/duchess/stuff
```

Создание новой файловой системы Ext4 выглядит так:

```
$ sudo mkfs.ext4 -L 'mylabel' /dev/sdb1
mke2fs 1.44.1 (24-Mar-2018)
/dev/sdb1 contains a XFS file system labelled 'stuff'
        created on Sun Sep 20 19:37:43 2020
Proceed anyway? (y,N) y
Creating filesystem with 466432 4k blocks and 116640 inodes
Filesystem UUID: 99da2e5d-f96a-4fb6-990d-599cf56247a2
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912

Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```

Кроме того, можно создать новый раздел и новую файловую систему на нем; см. примеры создания новых разделов в рецептах 8.4 и 9.4.

Комментарий

Создание новой файловой системы поверх существующей уничтожит все находящиеся в ней данные.

Параметр `-L` определяет метку тома. Меткой может быть любая строка длиной до 16 символов (FAT32 ограничивает метку 11 символами). Метки не являются обязательными, но могут пригодиться в некоторых случаях, например, их можно использовать в файле `/etc/fstab` вместо длинных UUID.

Параметр `-n` выполняет пробный прогон. Добавив его, вы увидите, что произойдет, без фактического создания новой файловой системы.

Команда `mke2fs` имеет множество параметров, но обычно используется только часть из них: имя устройства, метка тома, пробный запуск и создание внешнего журнала. Настройки по умолчанию для этой команды определяются в `/etc/mke2fs.conf`, и я советую не изменять их без тщательного изучения.

Дополнительная информация

- `man 8 mke2fs`
- Рецепт 8.4.
- Рецепт 11.5.

11.8. Настройка режима журналирования Ext4

Задача

Как известно, по умолчанию в Ext4 используется режим журналирования `data=ordered`, в котором журналируются не данные, а только метаданные. Этот режим обеспечивает хороший баланс между безопасностью и производительностью, но вы решили использовать самый безопасный режим `data=journal`.

Решение

Используйте команду `tune2fs`. Сначала проверьте текущий режим журналирования с помощью команды `dmesg`. Перед проверкой файловая система должна быть смонтирована:

```
$ dmesg | grep sdb1
[25023.525279] EXT4-fs (sdb1): mounted filesystem with ordered data mode.
```

Этот результат подтверждает, что в разделе /dev/sdb1 находится файловая система Ext4 и для нее установлен режим журналирования по умолчанию data=ordered. Теперь измените его на data=journal:

```
$ sudo tune2fs -o journal_data /dev/sdb1
tune2fs 1.44.1 (24-Mar-2018)
```

Размонтируйте и снова смонтируйте файловую систему и проверьте режим с помощью команды dmesg:

```
$ dmesg | grep sdb1
[25023.525279] EXT4-fs (sdb1): mounted filesystem with ordered data mode.
```

Если вы увидите множество строк с противоречивой информацией, например:

```
[ 206.076123] EXT4-fs (sdb1): mounted filesystem with journalled data mode.
[ 206.076433] EXT4-fs (sdb1): mounted filesystem with ordered data mode.
```

то перезагрузите систему, после чего должно остаться только сообщение mounted filesystem with journalled data mode (файловая система смонтирована в режиме журналирования данных).

Комментарий

Параметры, определяющие режим журналирования, в разной документации называются по-разному. В руководстве к man 8 tune2fs перечислены следующие параметры:

- journal_data;
- journal_data_ordered;
- journal_data_writeback.

В документации ядра и в различных инструкциях перечисляются параметры:

- data=journal;
- data=ordered;
- data=writeback.

Параметры data= предназначены для передачи ядру при загрузке либо через конфигурацию загрузчика, либо через файл /etc/fstab. Я предпочитаю простую и быструю команду tune2fs, которая работает со всеми файловыми системами Ext4, независимо от параметров монтирования.

Ниже представлены режимы журналирования в порядке уменьшения безопасности данных:

- `data=journal` — обеспечивает максимальную защиту данных. Все данные и метаданные сначала записываются в журнал, а затем в файловую систему. В случае сбоя этот режим дает наилучшие шансы на восстановление данных. Однако он также является самым ресурсоемким, поскольку все изменения записываются дважды;
- `data=ordered` — в этом режиме данные не записываются в журнал. Данные сначала записываются в файловую систему, а затем метаданные записываются в журнал. Метаданные логически сгруппированы по порядку и хранятся в одной транзакции. Когда они записываются на диск, сначала записываются связанные с ними блоки данных;
- `data=writeback` — самый быстрый и наименее безопасный режим. Данные сначала записываются в файловую систему, а затем метаданные записываются в журнал. Порядок записи данных не сохраняется. Я не думаю, что небольшой прирост производительности стоит дополнительного риска.

Дополнительная информация

- `man 8 tune2fs`
- Описание файловой системы Ext4 в документации ядра (https://oreil.ly/Y4ajq).

11.9. Определение журнала, к которому подключена файловая система Ext4

Задача

Есть несколько файловых систем Ext4. Часть из них с внутренними журналами, а часть — с внешними, и требуется узнать, какие журналы они используют.

Решение

Встречайте новую команду `dumpe2fs`. Она входит в набор e2fsprogs утилит ext2/3/4. Опросить файловую систему Ext4 с ее помощью можно следующим образом:

```
$ sudo dumpe2fs -h /dev/sda1 | grep -i uuid
dumpe2fs 1.43.8 (1-Jan-2018)
Filesystem UUID:          8593f3b7-4b7b-4da7-bf4a-cc6b0551cff8
Journal UUID:             f8e42703-94eb-49af-a94c-966e5b40e756
```

Journal UUID принадлежит журналу. Выполните команду `lsblk`, чтобы получить подробности:

```
$ lsblk -f | grep  f8e42703-94eb-49af-a94c-966e5b40e756
└─sdb5 ext4    journal1 f8e42703-94eb-49af-a94c-966e5b40e756
```

И вот она, файловая система Ext4, использующая внутренний журнал: в информации о ней отсутствует строка с UUID журнала:

```
$ sudo dumpe2fs -h /dev/sda2 | grep UUID
dumpe2fs 1.44.1 (24-Mar-2018)
Filesystem UUID:          64bfb5a8-0ef6-418a-bb44-6c389514ecfc
```

Комментарий

В Linux всегда есть какой-то способ узнать, где что находится. Команда `dumpe2fs` сообщает много полезной информации о файловых системах Ext4, включая UUID, время создания файловой системы, общее количество блоков, количество свободных блоков, размер журнала и многое другое.

Дополнительная информация

- `man 8 dumpe2fs`

11.10. Увеличение производительности Ext4 за счет использования внешнего журнала

Задача

Вы слышали, что если расположить журнал Ext4 на другом диске, то производительность файловой системы увеличится, и решили сделать это.

Решение

Внешний журнал действительно повышает производительность в режиме журналирования `data=journal`. (См. ниже подраздел «Комментарий», в котором приводится дополнительная информация о режимах журналирования.) Вы можете создать новую файловую систему Ext4 и внешний журнал или настроить существующую ФС для использования внешнего журнала.

Диски с файловой системой и журналом должны быть на одной машине и иметь одинаковую скорость чтения и записи. Если диск журнала будет работать медленнее, чем диск с ФС, то вы не получите заметного прироста производительности, если он вообще будет. Вы можете использовать два одинаковых твердотельных диска (SSD), два одинаковых жестких диска (HDD) или небольшой SSD для журнала и большой HDD для файловой системы, поскольку SSD намного быстрее, чем HDD.

Размещение журнала Ext4 на отдельном диске выполняется в несколько шагов. В следующем примере создаются два новых раздела: один для журнала, а другой — для новой файловой системы Ext4. Затем создается сам журнал, а потом ФС, после чего она подключается к журналу.

Первый раздел `/dev/sdb5` размером 200 Гбайт предназначен для журнала, второй раздел `/dev/sda1` размером 500 Гбайт предназначен для файловой системы Ext4:

```
$ sudo parted
(parted) select /dev/sdb
Using /dev/sdb
(parted) mkpart "journal1" ext4 1600GB 1800GB
(parted) select /dev/sda
Using /dev/sda
(parted) mkpart "ext4fs" ext4 1MB 500GB
```

Внешний журнал и файловая система должны иметь одинаковый размер блока, который в следующем примере задается с помощью параметра `-b 4096`. Если размер неизвестен, то определите его с помощью команды `tune2fs`. Следующие команды выполняются в командной оболочке Bash, а не в `parted`:

```
$ sudo tune2fs -l /dev/sda1  | grep -i 'block size'
Block size:               4096
```

Теперь создаем журнал (на это может потребоваться несколько минут) и потом новую файловую систему:

```
$ sudo mke2fs -b 4096 -O journal_dev /dev/sdb5
mke2fs 1.43.8 (1-Jan-2018)
/dev/sdb2 contains a ext4 file system labelled 'ext4'
        created on Mon Jan  4 18:25:30 2021
Proceed anyway? (y,N) y
Creating filesystem with 48747520 4k blocks and 0 inodes
Filesystem UUID: f8e42703-94eb-49af-a94c-966e5b40e756
Superblock backups stored on blocks:
Zeroing journal device:

$ sudo mkfs.ext4 -b 4096 -J device=/dev/sdb5 /dev/sda1
mke2fs 1.43.8 (1-Jan-2018)
Creating filesystem with 35253504 4k blocks and 8814592 inodes
Filesystem UUID: 8593f3b7-4b7b-4da7-bf4a-cc6b0551cff8
```

```
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872

Allocating group tables: done
Writing inode tables: done
Adding journal to device /dev/sdb2: done
Writing superblocks and filesystem accounting information: done
```

По завершении можно использовать новую файловую систему.

Подключить внешний журнал к существующей ФС можно с помощью команды `tune2fs`. Сначала очистите журнал в существующей файловой системе, а затем свяжите ее с внешним журналом:

```
$ sudo tune2fs -O ^has_journal /dev/sda1
$ sudo tune2fs -b 4096 -J device=/dev/sdb5 /dev/sda1
```

Комментарий

Журнал Ext4 обеспечивает дополнительную защиту данных в случае сбоя диска или системы, отслеживая изменения, которые еще не записаны на диск. Даже если во время сбоя потеряются ваши самые последние изменения, журнал защитит файловую систему от повреждения, то есть вы потеряете только малую часть, но не всю работу.

Перенос журнала на отдельный диск на том же компьютере дает заметное увеличение производительности в режиме журналирования `data=journal`. В Ext4 есть три режима журналирования: `journal`, `ordered` и `writeback`. По умолчанию используется режим `ordered`. См. рецепт 11.8, в котором описываются подробности этих режимов, которые помогут выбрать самый подходящий для вас.

Символ каретки ^ отключает функцию. В примере выше он очищает существующий внутренний журнал.

Журналы Ext4 могут использоваться только одной файловой системой.

Дополнительная информация

- `man 8 mke2fs`
- `man 8 tune2fs`
- Глава 8.
- Глава 9.

11.11. Освобождение пространства, занятого зарезервированными блоками, в файловой системе Ext4

Задача

Большинство дистрибутивов Linux резервируют 5 % пространства файловых систем Ext4 для пользователя root и системных служб. На больших современных жестких дисках это довольно много, и было бы желательно освободить часть этого места.

Решение

Используйте команду `tune2fs` для изменения размера свободного пространства в файловой системе Ext4. Указать размер можно в процентах, например, команда в следующем примере уменьшит его до 1 %:

```
$ sudo tune2fs -m 1 /dev/sda1
tune2fs 1.44.1 (24-Mar-2018)
Setting reserved blocks percentage to 1% (820474 blocks)
```

Это все еще около 3 Гбайт при размере блока 4 Кбайт (820 474 × 4096 = = 3 360 661 504 байт). Узнать размер блока можно следующим образом:

```
$ sudo tune2fs -l /dev/sda1  | grep -i 'block size'
Block size:               4096
```

При желании можно задать размер в долях процента:

```
$ sudo tune2fs -m .25 /dev/sda1
tune2fs 1.44.1 (24-Mar-2018)
Setting reserved blocks percentage to 0.25% (205118 blocks)
```

Это примерно 800 Мбайт.

Кроме того, можно задать размер в блоках:

```
$ sudo tune2fs -r 250000 /dev/sda1
tune2fs 1.44.1 (24-Mar-2018)
Setting reserved blocks count to 250000
```

Двести пятьдесят тысяч блоков по 4 Кбайт в каждом — это около гигабайта. Проверьте результат:

```
$ sudo tune2fs -l /dev/sda1 | grep -i 'reserved block'
Reserved block count:     250000
```

Комментарий

Даже если вы исчерпаете все свободное место на диске, то все равно сможете войти в систему как root и удалить лишнее, чего вы не смогли бы сделать, если бы эти 5 % не были зарезервированы. Однако 5 % — это пережиток времен жестких дисков небольшой емкости. Жесткие диски теперь способны вместить так много, что нет необходимости резервировать целых 5 %. Например, 5 % от емкости диска 1 Тбайт — около 50 Гбайт. Достаточно зарезервировать несколько сотен мегабайт. Я у себя резервирую 1 Гбайт. Эту цифру легко запомнить, и такого объема вполне достаточно.

Используйте команду `dumpe2fs`, чтобы проверить настроенное количество зарезервированных блоков в вашей файловой системе Ext4:

```
$ sudo dumpe2fs -h /dev/sda1
[...]
Block count:              82047488
Reserved block count:     250000
[...]
```

Дополнительная информация

- `man 8 dumpe2fs`
- `man 8 tune2fs`

11.12. Создание новой файловой системы XFS

Задача

Вам нравится XFS, и вы хотели бы создать новую файловую систему XFS.

Решение

Для этого нужно установить пакет *xfsprogs* и выделить раздел для новой файловой системы. После этого можно создать новую файловую систему XFS с по-

мощью команды mkfs.xfs. Следующий пример демонстрирует все эти шаги. Здесь используется раздел /dev/sda1, и новая файловая система получает метку xfstest:

```
$ sudo apt install xfsprogs
$ sudo parted /dev/sda mkpart testxfs xfs 1MB 500GB
$ sudo mkfs.xfs -L xfstest /dev/sda1
meta-data=/dev/sdb5              isize=512    agcount=4, agsize=640000 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=0, rmapbt=0,
reflink=0
data     =                       bsize=4096   blocks=2560000, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
```

Проверяем получившийся результат с помощью команды lsblk:

```
$ lsblk -f | grep -w sda1
├─sda1 xfs xfstest bb5dddb3-af74-4bed-9d2a-e79589278e84
```

Смонтируйте новую файловую систему, настройте атрибуты владения и разрешения, и она готова к применению. Следующий пример монтирует ее в /mnt/xfstest, устанавливает владельцем пользователя Duchess, устанавливает разрешения на чтение и запись для Duchess и только на чтение для всех остальных:

```
$ sudo mkdir /mnt/xfstest
$ sudo mount /dev/sda1 /mnt/xfstest
$ sudo chown -R duchess:duchess /mnt/xfstest
$ sudo chmod -R -755 /mnt/xfstest
```

Комментарий

Команда, создающая новую файловую систему XFS, выводит много полезной информации, например размер блока, количество блоков и размер сектора.

Дополнительная информация

- man 8 mkfs.xfs

11.13. Изменение размера файловой системы XFS

Задача

Изменить размер файловой системы XFS.

Решение

Размер файловой системы XFS можно только увеличить. Чтобы уменьшить ее размер, придется скопировать данные куда-то еще, создать раздел меньшего размера, отформатировать его в XFS и затем скопировать данные обратно.

Увеличить размер значительно проще. Для этого в конце раздела, в котором находится файловая система XFS, должно иметься свободное пространство. В следующих примерах новая конечная точка раздела — 2700 Гбайт, а файловая система смонтирована в `/media/duchess/xfs`.

Запустите `parted`. Выведите информацию о разделе, чтобы убедиться в правильном выборе раздела и конечной точки, увеличьте размер раздела и выйдите из `parted`:

```
$ sudo parted /dev/sdb
GNU Parted 3.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p free
Model: ATA SAMSUNG HD204UI (scsi)
Disk /dev/sdb: 4000GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name    Flags
        17.4kB  1049kB  1031kB  Free Space
 1      1049kB  1656GB  1656GB  xfs          files
 2      1656GB  1759GB  103GB   xfs          files2
        1759GB  4000GB  242GB   Free Space

(parted) resizepart 2
(parted) Warning: Partition /dev/sdb2 is being used. Are you sure you want to
continue?
Yes/No? Yes
End?  [1759GB]? 1900GB
(parted) q
```

Теперь можно увеличить размер файловой системы, чтобы она заняла раздел полностью:

```
$ sudo xfs_growfs /media/duchess/xfs
```

Вот и все! Наслаждайтесь увеличенным размером файловой системы.

Комментарий

Кроме того, можно размонтировать файловую систему и изменить ее размер в автономном режиме. Это немного безопаснее.

С помощью GParted вы сможете изменить размер файловой системы еще быстрее и проще (см. рецепт 9.7).

Дополнительная информация

- Рецепт 8.8.
- Рецепт 9.7.

11.14. Создание файловой системы exFAT

Задача

Флеш-накопитель вашей цифровой камеры отформатирован в файловой системе exFAT, или у вас есть другие устройства флеш-памяти с данной файловой системой и вы хотите иметь возможность читать, записывать и редактировать файлы на носителях с этих устройств в системе Linux.

Решение

Есть два возможных решения этой задачи: применить реализацию exFAT, работающую в пространстве пользователя (Filesystem in Userspace, FUSE), или встроенную реализацию, работающую в ядре Linux. В данном рецепте мы задействуем exFAT FUSE, поскольку на момент написания этих строк встроенная реализация была включена не во все дистрибутивы. Найдите ядро версии 5.7 и ознакомьтесь с примечаниями к выпуску, а также с новостями вашего дистрибутива. (Выполните команду `uname -r`, чтобы узнать версию ядра.)

Имена пакетов с реализацией exFAT могут отличаться в разных дистрибутивах. Пакеты exfat-fuse и exfat-utils — старые, exfatprogs — новейшая реализация,

заменяющая как exfat-fuse, так и exfat-utils. Если вам доступна только старая версия — устанавливайте хотя бы ее.

Команда создания новой файловой системы exFAT одинакова для обоих пакетов. Следующий пример форматирует `/dev/sdc1` в exFAT:

```
$ sudo mkfs.exfat /dev/sdc1
mkexfatfs 1.2.8
Creating... done.
Flushing... done.
File system created successfully.
```

exFAT специально создавалась максимально простой, поэтому параметров создания не так много. Назначить метку можно следующим образом:

```
$ sudo exfatlabel /dev/sdc2 exfatfs
```

Проверьте получившийся результат с помощью команды `lsblk`:

```
$ lsblk -f
NAME   FSTYPE LABEL   UUID
sdc
├─sdc1
├─sdc2 exfat  exfatfs 8178-51D4
└─sdc3
```

Комментарий

Вам не нужно создавать специальный раздел exFAT для чтения файлов из exFAT на других устройствах, достаточно установить пакеты поддержки exFAT.

Если для разметки диска вы предпочитаете использовать инструмент с графическим интерфейсом, то сразу отмечу, что GParted не поддерживает exFAT по юридическим причинам. Однако эта файловая система поддерживает в GNOME Disks. Вам не нужно устанавливать GNOME, чтобы воспользоваться данной программой; просто установите пакет *gnome-disk-utility*.

Microsoft выпустила спецификацию exFAT в 2019 году. А Samsung выпустила *exfatprogs* в начале 2020 года. К тому времени, когда вы будете читать эти строки, встроенная поддержка exFAT должна появиться в последних выпусках Fedora, Ubuntu и openSUSE Tumbleweed.

Дополнительная информация

- `man 8 exfat`
- `man 8 exfatlabel`

11.15. Создание файловых систем FAT16 и FAT32

Задача

Узнать, как создать файловую систему FAT16 или FAT32.

Решение

Вам понадобится пакет *dosfstools*, который по умолчанию устанавливается в большинстве дистрибутивов Linux. Следующие примеры демонстрируют создание нового раздела размером 500 Мбайт с помощью `parted` и его форматирование в FAT32.

Создайте новый раздел и обратите внимание, как в примере меняется единица измерения на `mb` (мегабайты) и как используется `mkpart` в интерактивном режиме:

```
$ sudo parted /dev/sdb
GNU Parted 3.2
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Model: ATA SAMSUNG HD204UI (scsi)
Disk /dev/sdb: 2000399MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start    End     Size    File system  Name   Flags
1       0.00GB   1656GB  1656GB  xfs          files

(parted) unit mb
mkpart
Partition name?  []?
File system type?  [ext2]? fat32
Start? 1656331MB
End? 1656831MB
(parted) print
Model: ATA SAMSUNG HD204UI (scsi)
Disk /dev/sdb: 2000399MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start      End        Size       File system  Name  Flags
 1      1.05MB     1656331MB  1656330MB  xfs          bup
 2      1656331MB  1656831MB  500MB      fat32

(parted) q
```

Указывать имя раздела (в запросе `Partition name?`) не обязательно; в этом примере я оставила имя раздела пустым. Теперь создайте новую файловую систему FAT32:

```
$ sudo mkfs.fat -F 32 -n fat32test /dev/sdb2
mkfs.fat 4.1 (2017-01-24)
mkfs.fat: warning - lowercase labels might not work properly with DOS or Windows
```

Выполните проверку с помощью команды `lsblk`:

```
$ lsblk  -f /dev/sdb
NAME   FSTYPE LABEL       UUID                                 FSAVAIL FSUSE%
MOUNTPOINT
sdb
├─sdb1 xfs    xfstest     1d742b2d-a621-4454-b4d3-469216a6f01e
└─sdb2 vfat   fat32test   AB39-1808
```

Комментарий

Чтобы создать файловую систему FAT16, используйте параметр `-F 16`.

Файлы в FAT16 и сама эта файловая система не могут иметь размер больше 4 Гбайт.

В FAT32 максимальный размер файлов ограничен 4 Гбайт, а размер раздела — 16 Тбайт при использовании секторов размером 4 Кбайт и кластеров 64 Кбайт.

Дополнительная информация

- Глава 8.
- Глава 9.
- `man 8 mkfs.fat`

11.16. Создание файловой системы Btrfs

Задача

Btrfs — отличная файловая система и достойна того, чтобы ее попробовать.

Решение

Это отличная и в то же время очень сложная файловая система. SUSE Linux Enterprise Server (SLES) и openSUSE — лучшие дистрибутивы Linux для опробования Btrfs. Проекты SLES и openSUSE — крупнейшие разработчики Btrfs, и они создали отличный инструмент Snapper для управления моментальными

снимками Btrfs. Они также предоставляют самую полную документацию. Разметка диска по умолчанию в openSUSE/SLES предполагает настройку подтомов Btrfs и автоматическое создание моментальных снимков.

Для начала скачайте последнюю версию openSUSE Tumbleweed. Запустите установщик и, когда откроется экран Suggested Partitioning (Предлагаемая разметка), взгляните на первое предложение (рис. 11.2).

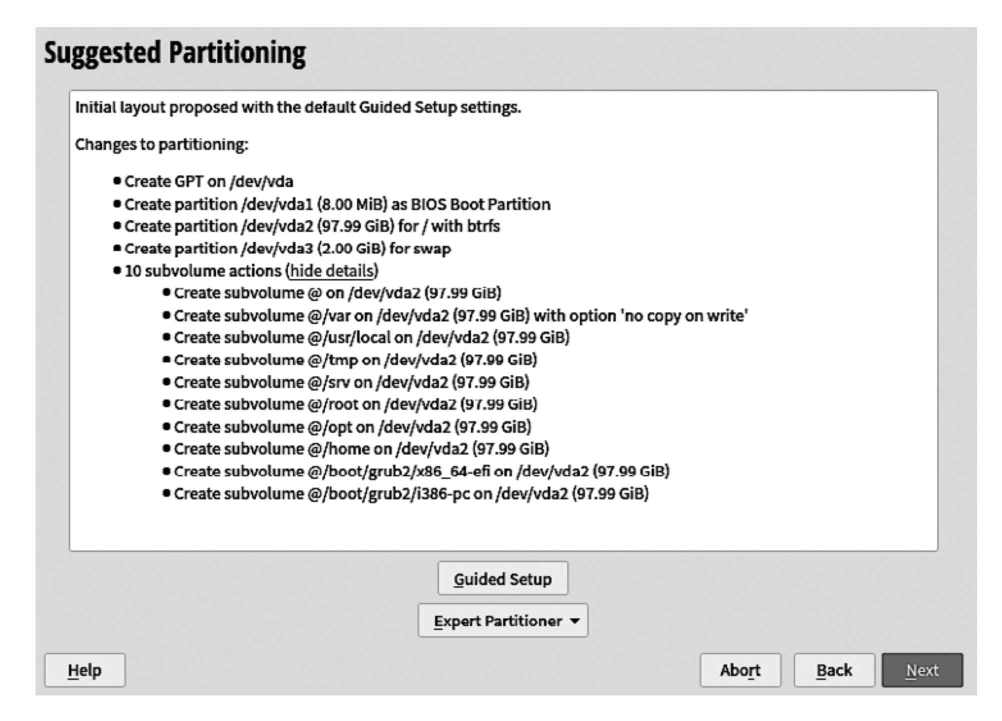


Рис. 11.2. Первое предложение по разметке диска в процедуре установки openSUSE

Нажмите кнопку Guided Setup (Мастер установки), чтобы изменить это предложение. Пропустите экран Enable logical volume management (LVM) / Enable disk encryption (Включить управление логическими томами (LVM)/Включить шифрование диска) и остановитесь на экране Filesystem Options (Параметры файловой системы). Выберите Propose Separate Home Partition (Предложить отдельный домашний раздел) и отформатируйте его как Btrfs.

Установите оба флажка для Propose Separate Swap Partition (Предложить отдельный раздел подкачки), затем нажмите кнопку Next (Далее) (рис. 11.3).

После этого вы вернетесь на экран Suggested Partitioning (Предлагаемая разметка). Если вы решите изменить размеры разделов, то нажмите кнопку Expert Partitioner

(Экспертная разметка) и выберите Start with Current Proposal (Начать с текущего предложения) (рис. 11.4). Иначе нажмите кнопку Next (Далее) и продолжите установку.

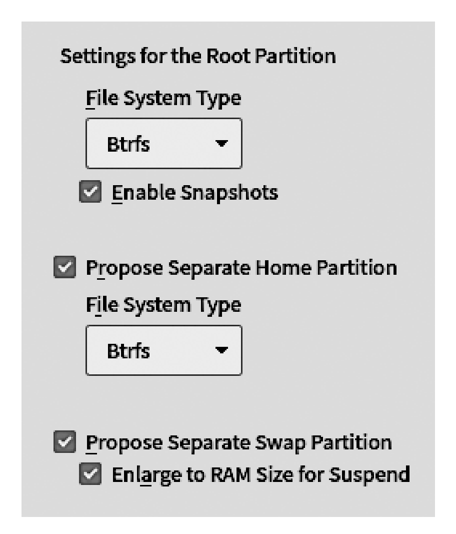


Рис. 11.3. Создать домашний раздел

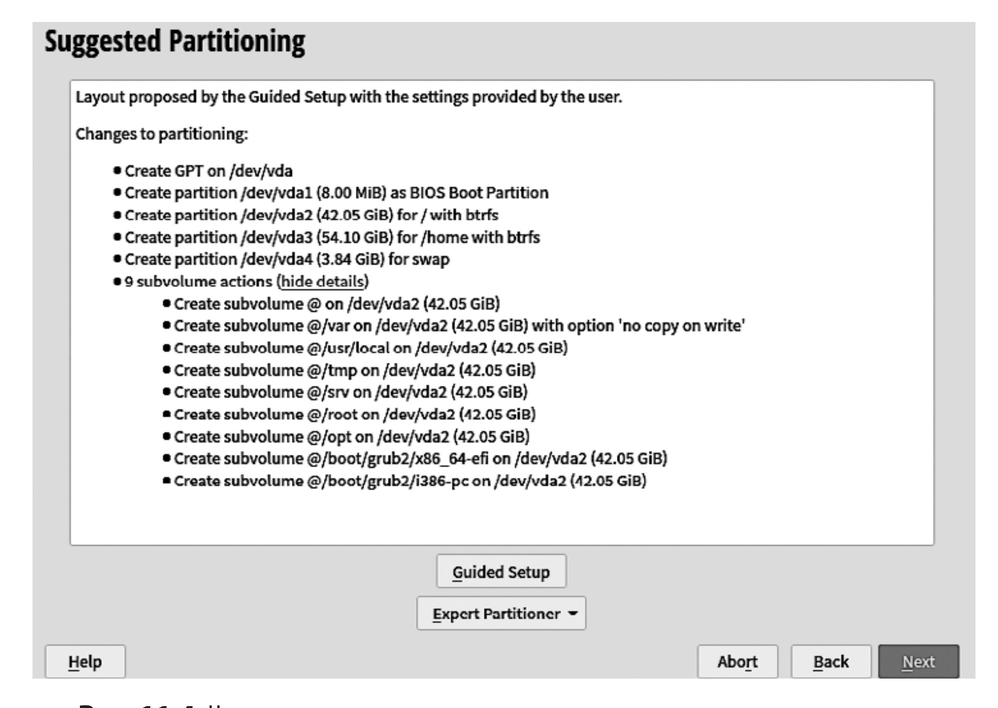


Рис. 11.4. Нестандартная разметка диска с использованием начального предложения

По завершении у вас будет готовая к использованию система Linux с Btrfs, уже настроенная хорошими значениями по умолчанию.

Комментарий

Настройка Btrfs вручную — довольно сложная задача, хотя когда вы познакомитесь с этой файловой системой поближе, то можете попробовать настроить его вручную. Мне нравится узнавать что-то новое, начиная с работающей реализации. Однако я не смогу уместить полезные инструкции по настройке Btrfs в несколько рецептов. Btrfs настолько гибкая и богатая возможностями файловая система, что для ее описания нужна отдельная книга. И такая книга есть, она написана трудолюбивыми сотрудниками SUSE. Для начала прочитайте начальное руководство Startup Guide, в котором описывается порядок установки, и затем прочитайте раздел о восстановлении системы и управление снимками с помощью Snapper под названием System Recovery and Snapshot Management with Snapper в документации openSUSE (https://oreil.ly/1Vi9L). Snapper и Btrfs — отличное сочетание для управления Btrfs и быстрого восстановления после сбоев.

Дополнительная информация

- Начальные и справочные руководства в документации к openSUSE (https://oreil.ly/1Vi9L).
- Руководства по развертыванию и администрированию в SLES Product Manuals (https://oreil.ly/fX5G9).

ГЛАВА 12

Безопасный удаленный доступ с OpenSSH

OpenSSH — популярный инструмент для безопасного удаленного администрирования. Он шифрует весь трафик, передаваемый во время сеанса, и гарантирует целостность передачи данных. Если что-то или кто-то подменит ваши пакеты, то SSH сообщит вам об этом. В данной главе вы узнаете, как настроить доступ к удаленным хостам через SSH, управлять своими ключами шифрования, настроить свои учетные данные на нескольких удаленных хостах, настроить приглашение Bash, чтобы отличать сеансы SSH, и многое другое.

OpenSSH поддерживает большое количество надежных алгоритмов шифрования, и ни один из них не обременен патентами, поскольку команда OpenSSH приложила большие усилия, чтобы гарантировать, что внутри OpenSSH не будет запатентованного или иным образом обремененного кода. В рецепте 12.16 вы увидите, как вывести список всех поддерживаемых алгоритмов.

OpenSSH — это набор утилит для передачи данных по сети:

- *sshd* — демон сервера OpenSSH;
- *ssh* — сокращенно от secure shell («безопасная оболочка»), но на самом деле не включает оболочку, а обеспечивает безопасный канал связи с командной оболочкой в удаленной системе;
- *scp* — сокращенно от secure copy («безопасное копирование»), предназначена для шифрованной передачи файлов;
- *sftp*, secure file transfer protocol («протокол безопасной передачи файлов»), обеспечивает доступ к файлам;
- *ssh-copy-id* — замечательная маленькая программа для сохранения вашего открытого ключа в файле `authorized_keys` на удаленном сервере SSH;

- *ssh-keyscan* — отыскивает и собирает открытые ключи хостов в сети, избавляя от необходимости искать их вручную;
- *ssh-keygen* — генерирует ключи аутентификации и управляет ими;
- *ssh-add* — добавляет вашу идентификационную информацию в настройки агента аутентификации *ssh-agent*.

В этой главе вы познакомитесь с ssh, sshd, ssh-copy-id, ssh-keygen и двумя полезными утилитами: sshfs и ssh-agent.

Утилита sshfs монтирует удаленные файловые системы в каталог на локальном компьютере, а ssh-agent запоминает парольные фразы с вашими личными ключами SSH для автоматической аутентификации. ssh-agent привязывается к одному сеансу входа в систему, поэтому выход из системы или открытие другого терминала предполагает запуск агента заново. Лучшей утилитой для автоматизации операций является Keychain — интерфейс для ssh-agent. Keychain автоматически использует ssh-agent, пока вы не перезагрузите компьютер, поэтому вводить парольные фразы вам потребуется только при запуске (см. рецепт 12.10).

OpenSSH поддерживает разные типы аутентификации, такие как:

- *аутентификация с паролем* — выполняется с помощью ваших имени пользователя и пароля. Это самый простой и гибкий вариант, поскольку позволяет войти в систему с любого компьютера. Но вы должны проявить разумную осторожность и не открывать сеансы SSH с ненадежных компьютеров, например установленных в библиотеке или интернет-кафе. Если на этот компьютер будет установлена программа-кейлоггер (клавиатурный шпион), то она перехватит ваши учетные данные;
- *аутентификация с открытым ключом* — это аутентификация с применением личных открытых ключей SSH без использования учетных данных. Этот способ более хлопотный, поскольку вам придется создавать и распространять ваши открытые ключи, а кроме того, вы сможете входить в систему только с компьютеров, на которых хранится ваш закрытый ключ. Ряд коммерческих служб требуют от клиентов использования той или иной формы аутентификации с открытым ключом;
- *аутентификация без парольной фразы* — аутентификация с открытым ключом без парольной фразы. Удобный способ для средств автоматизации, таких как сценарии и задания cron. Однако любой, кому удастся украсть ваш закрытый ключ, сможет замаскироваться под вас, поэтому вы должны очень тщательно защищать свой закрытый ключ, используемый для аутентификации без парольной фразы.

Альтернативой использованию ключей без парольных фраз является Keychain, который запомнит ваши личные ключи за вас (см. рецепт 12.10).

Есть два разных типа ключей аутентификации: ключи хоста, которые аутентифицируют компьютеры, и открытые ключи, аутентифицирующие пользователей. Ключи SSH создаются парами, состоящими из закрытого и открытого ключа. Открытый ключ служит для шифрования информации перед отправкой в сеть, а закрытый — для расшифровывания, что является гениально простой схемой. Вы можете безопасно распространять свои открытые ключи сколько хотите, но должны защищать свой закрытый ключ и не передавать его никому.

Сервер и клиент определяются направлением транзакции. На сервере запущен демон SSH, и он принимает запросы на соединение, а клиент — это любой входящий на этот компьютер через SSH.

12.1. Установка сервера OpenSSH

Задача

Установить сервер OpenSSH.

Решение

В большинстве дистрибутивов Linux клиент OpenSSH устанавливается по умолчанию, а сервер — не всегда. Пакеты с OpenSSH в разных дистрибутивах Linux называются по-разному, поэтому используйте диспетчер пакетов, чтобы отыскать нужный пакет для вашего дистрибутива (см. приложение). Установите сервер, затем проверьте, запустился ли он:

```
$ systemctl status sshd
● sshd.service - OpenSSH Daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; disabled; vendor preset
   Active: inactive (dead)
   [...]
```

В данном примере сервер не работает и не включен. В большинстве дистрибутивов Linux не предполагается автоматический запуск сервера OpenSSH после установки. И в общем случае это верное решение, поскольку сервер нужно правильно настроить, прежде чем позволить ему принимать запросы на подключение. Если сервер запустился до того, как вы проверили конфигурацию, то остановите его или заблокируйте его порты с помощью брандмауэра.

Следующие шаги — настройка ключей шифрования и сервера, которые описываются в рецептах 12.2 и 12.3.

Комментарий

Помните, что, используя термины «сервер» и «клиент», мы подразумеваем не только оборудование, но и направление транзакции. Роль сервера выполняет компьютер, на котором запущен демон SSH, принимающий запросы на соединение, а роль клиента — любой компьютер, обращающийся к серверу через SSH. Любой персональный компьютер с Linux может быть сервером, клиентом или и тем и другим.

Дополнительная информация

- Глава 14.
- Сайт OpenSSH (https://openssh.com).
- sshd (8).
- Приложение (в конце книги).

12.2. Генерирование новых ключей хоста

Задача

Сгенерировать ключи хоста, поскольку ваш дистрибутив Linux не создает их автоматически при установке, или вы решили заменить существующие ключи, или вы клонируете установленную систему или виртуальную машину и для клонов нужно создать свои уникальные ключи.

Решение

Используйте команду `ssh-keygen`. Существует несколько типов ключей: RSA, DSA, ECDSA и ED25519. Сначала удалите старые ключи, если они имеются:

```
$ sudo rm /etc/ssh/ssh_host*
```

Затем создайте новые:

```
$ sudo ssh-keygen -A
ssh-keygen: generating new host keys: RSA DSA ECDSA ED25519
```

Комментарий

Если вам когда-нибудь станет скучно и захочется чем-нибудь заняться, то попробуйте изучить вопрос: «Какие форматы ключей SSH лучше использовать?» Споры на эту тему бесконечны. Но я могу дать короткий ответ: используйте RSA, ECDSA и ED25519 и избегайте DSA. Удалите DSA-ключ хоста, а остальные оставьте.

RSA — самый старый тип. Ключи этого типа достаточно стойкие и обеспечивают максимальную совместимость.

ECDSA и ED25519 — более новые типы ключей, очень эффективные и менее затратные в вычислительном отношении.

Некоторые старые клиенты SSH не поддерживают ECDSA и ED25519. Надеюсь, вы не используете их, поскольку ECDSA и ED25519 появились вместе с OpenSSH 6.5 в 2014 году. Чрезвычайно важно обновлять службы безопасности и не использовать старые и небезопасные клиенты.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- ssh-keygen (1).

12.3. Настройка сервера OpenSSH

Задача

Настроить сервер OpenSSH на максимальный уровень безопасности и протестировать его.

Решение

Для начала убедитесь, что закрытые ключи хоста вашего сервера доступны только root и только для чтения:

```
$ ls -l /etc/ssh/
-r-------- 1 root root    227 Jun  4 11:30 ssh_host_ecdsa_key
-r-------- 1 root root    399 Jun  4 11:30 ssh_host_ed25519_key
-r-------- 1 root root   1679 Jun  4 11:30 ssh_host_rsa_key
```

Они должны выглядеть так, как показано в коде выше. Затем проверьте свои открытые ключи, принадлежащие пользователю root, они должны быть доступны для чтения и записи владельцу и только для чтения всем остальным:

```
$ ls -l /etc/ssh/
-rw-r--r-- 1 root root    174 Jun  4 11:30 ssh_host_ecdsa_key.pub
-rw-r--r-- 1 root root     94 Jun  4 11:30 ssh_host_ed25519_key.pub
-rw-r--r-- 1 root root    394 Jun  4 11:30 ssh_host_rsa_key.pub
```

Вот так правильно.

Теперь несколько слов о конфигурационном файле `/etc/ssh/sshd_config`. Изменив его содержимое, перезагрузите sshd, чтобы загрузить ваши изменения:

```
$ sudo systemctl reload sshd.server
```

Раскомментируйте настройки в этом файле, которые хотите изменить.

Настройте sshd для проверки правильности атрибутов файлов и домашних каталогов пользователей, прежде чем принимать от них запросы на подключение:

```
StrictModes yes
```

Если разрешения для файлов настроены неверно, то этот параметр не позволит им войти в систему.

Если у вашего компьютера более одного IP-адреса, то определите, какой адрес или адреса будет прослушивать сервер:

```
ListenAddress 192.168.10.15
ListenAddress 192.168.10.16
```

Вы можете назначить нестандартные порты для приема соединений сервером sshd. Используйте только порты выше 1024 и загляните в `/etc/services`, чтобы выбрать неиспользуемые порты, затем добавьте новые порты в `/etc/services`:

```
sshd 2022
sshd 2023
```

После этого добавьте их в файл `/etc/ssh/sshd_config`:

```
Port 2022
Port 2023
```

Можно разрешить доступ только указанным группам (создайте эти группы в `/etc/group`):

```
AllowGroups webadmins backupadmins
```

Или отказать в доступе каким-то группам с помощью `DenyGroups`.

Отключите возможность входа с привилегиями root. Безопаснее входить в систему с привилегиями рядового пользователя и затем использовать `sudo`:

```
PermitRootLogin no
```

Как вариант, можно разрешить вход с привилегиями root, но только методом аутентификации с открытым ключом:

```
PermitRootLogin prohibit-password
```

Кроме того, можно отключить вход с паролем для всех пользователей и разрешить только аутентификацию с открытым ключом (см. рецепт 12.7):

```
PasswordAuthentication no
```

Можно запретить вход конкретным пользователям, перечислив их имена или имена хостов или IP-адреса:

```
DenyUsers duchess madmax stash@example.com cagney@192.168.10.25
```

Или разрешить вход ограниченному кругу пользователей, определив параметр `AllowUsers`. Вы можете задействовать оба параметра, но помните при этом, что `DenyUsers` всегда обрабатывается первым.

Ограничьте время ожидания, пока пользователь введет свои учетные данные и завершит процедуру входа в систему. По умолчанию 120 секунд:

```
LoginGraceTime 90
```

Можно также ограничить количество неудачных попыток входа. По умолчанию 6:

```
MaxAuthTries 4
```

Комментарий

Любой сканер портов найдет ваши открытые порты, и злоумышленники могут попытаться взломать пароль простым методом перебора. Злоумышленники выбирают для атаки SSH-порт 22 по умолчанию. Изменение номера порта не слишком снизит риски, но все же предотвратит какую-то часть атак. Прежде чем использовать альтернативные номера портов, загляните в файл `/etc/services`, найдите неиспользуемые порты, а затем впишите в этот файл порты, выбранные вами.

Аутентификация с открытым ключом очень надежна, и ее нельзя взломать, как аутентификацию с паролем (см. рецепт 12.7). Однако это менее удобный спо-

соб, поскольку позволяет входить в систему только с компьютеров, на которых имеется ваш закрытый ключ.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- `man 5 sshd_config`
- Рецепт 12.5.
- Рецепт 12.7.

12.4. Проверка синтаксиса конфигурации

Задача

Все люди делают ошибки, и вам нужно средство для проверки синтаксиса файла `/etc/ssh/sshd_config`.

Решение

И такое средство есть. После внесения изменений выполните следующую команду:

```
$ sudo sshd -t
```

Если в файле нет синтаксических ошибок, то команда просто завершится без вывода каких-либо сообщений. Если ошибки есть, то она сообщит о них:

```
$ sudo sshd -t
/etc/ssh/sshd_config: line 9: Bad configuration option: Porotocol
/etc/ssh/sshd_config: terminating, 1 bad configuration options
```

Такую проверку можно делать при работающем демоне SSH. Это позволит вам исправить свои ошибки перед вводом команды перезагрузки или перезапуска.

Комментарий

Параметр `-t` означает `test` (проверка). При запуске с этим параметром команда не затрагивает демон SSH и просто проверяет наличие синтаксических ошибок в файле `/etc/ssh/sshd_config`, так что команду `sshd` с параметром `-t` можно использовать в любой момент.

Дополнительная информация

- man 5 sshd_config
- Сайт OpenSSH (https://openssh.com).

12.5. Настройка аутентификации с паролем

Задача

Настроить клиент OpenSSH для входа на удаленный хост с использованием простейшего из поддерживаемых методов.

Решение

Самый простой метод удаленного доступа по SSH — аутентификация с паролем. Для его реализации потребуются:

- настроенный сервер OpenSSH на удаленной машине, на которую вы хотите войти (см. рецепт 12.3);
- запущенный демон sshd на этой удаленной машине и порт 22 или любой другой порт, который использует sshd, не должен блокироваться брандмауэрами;
- клиент SSH на вашем клиентском компьютере;
- наличие вашей собственной учетной записи пользователя на удаленной машине;
- ключи хоста на сервере (см. рецепт 12.2).

Открытый ключ хоста удаленной машины должен быть передан клиентам. Самый простой способ — войти в систему с клиента и позволить OpenSSH передать ключ:

```
duchess@pc:~$ ssh duchess@server1
  The authenticity of host 'server1 (192.168.43.74)' can't be established.
  ECDSA key fingerprint is SHA256:8iIg9wwFIzLgwiiQ62WNLF5oOS3SL/aTw6gFrtVJTx8.
  Are you sure you want to continue connecting (yes/no)? *yes*
  Warning: Permanently added 'server1,192.168.43.74' (ECDSA) to the list of
  known hosts.
  Password: password
  Last login: Wed Jul 8 19:22:39 2021 from 192.168.43.183
  Have a lot of fun...
```

Теперь пользователь Duchess может работать на `server1`, как если бы `server1` был локальной машиной. При этом весь трафик между клиентом и сервером будет шифроваться.

Обмен ключами хоста происходит только один раз, при первом входе в систему. Повторный обмен никогда не будет инициирован, если ключ не был заменен новым или вы не удалили его из своего личного файла `~/.ssh/known_hosts`.

Комментарий

Открытый ключ хоста `server1` хранится в файле `~/.ssh/known_hosts` на клиентском компьютере. Этот файл может содержать любое количество ключей хоста.

Входить в систему как root через SSH небезопасно; лучше выполнить вход с привилегиями рядового пользователя, а затем использовать `su` или `sudo`. Войти в систему можно с учетными данными любого пользователя, если вы знаете его пароль:

```
duchess@pc:~$ ssh madmax@server1
```

Если у вас одно и то же имя пользователя на обеих машинах, то его можно не указывать и входить в систему так:

```
duchess@pc:~$ ssh server1
```

У меня есть привычка всегда указывать имя пользователя как недорогую страховку от ошибок.

Пусть вас не смущают термины *«клиент»* и *«сервер»*. Речь не об оборудовании. Сервер — это компьютер, на который вы входите, а клиент — компьютер, откуда вы входите. На клиенте демон sshd может быть не запущен.

Есть риск, что при передаче ключ хоста будет перехвачен и заменен поддельным ключом, вследствие чего злоумышленник получит доступ к вашим системам. Вы можете проверить отпечаток (контрольную сумму) открытого ключа, прежде чем ввести **`yes`**. Для этого можно применить старый и надежный метод, такой как запись на листе бумаги и сравнение, или новомодный, например сфотографировать ключ хоста на телефон для сравнения или использовать свой телефон по прямому назначению и позвонить кому-то, у кого есть доступ к удаленному компьютеру, чтобы тот прочитал вам отпечаток ключа.

См. рецепт 12.6, чтобы узнать, как получить отпечаток ключа.

Дополнительная информация

- Рецепт 12.6.
- Сайт OpenSSH (https://openssh.com).
- `man 1 ssh`
- `man 1 ssh-keygen`
- `man 8 sshd`

12.6. Получение отпечатка ключа

Задача

Получить отпечаток ключа хоста для проверки его достоверности.

Решение

Выполните команду `ssh-keygen` на сервере, передав ей ключ хоста, отпечаток которого требуется получить:

```
duchess@server1:~$ ssh-keygen -lf /etc/ssh/ssh_host_rsa_key
4096 SHA256:32Pja4+F2+MTdla9cs4ucecThswRQp6a4xZ+5sC+Bf0 backup server1 (RSA)
```

Комментарий

Это тот случай, когда пригодятся старые добрые способы связи, такие как телефон. Не используйте электронную почту, если у вас не настроено ее шифрование, поскольку незашифрованную электронную почту легко перехватить и прочитать.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- `man 1 ssh-keygen`

12.7. Аутентификация с открытым ключом

Задача

Настроить аутентификацию с открытым ключом, поскольку она надежнее аутентификации с паролем и не использует ваш пароль Linux. Вам хотелось бы использовать один открытый ключ для доступа к нескольким системам или для каждой из них создать уникальный открытый ключ.

Решение

Да, пользователям Linux доступно все это. Вы можете создать столько ключей SSH, сколько захотите, и применять их как хотите. Ниже приводится мое любимое заклинание для создания новой пары ключей RSA. Конечно, вы можете

использовать собственный комментарий и название ключа. (См. ниже подраздел «Комментарий», чтобы узнать, нужно ли задавать парольную фразу для закрытого ключа.)

```
duchess@pc:~/.ssh $ ssh-keygen -C "backup server2" -f id-server2 -t rsa -b 4096
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id-server2.
Your public key has been saved in id-server2.pub.
The key fingerprint is:
SHA256:32Pja4+F2+MTdla9cs4ucecThswRQp6a4xZ+5sC+Bf0 backup server2
The key's randomart image is:
+---[RSA 4096]----+
|          ..     |
|          ....   |
|           o. . .|
|          +  .  o|
|        S* .o o o|
|        +.+..Bo*+|
|         *.+*EX=o|
|        o *o.Oo+.|
|         o.o=+*+.|
+----[SHA256]-----+
```

Следующий шаг — скопировать новенький ключ на удаленный компьютер, которым в данном случае является локальный сервер для хранения резервных копий `server1`. При этом у вас уже должен быть настроен SSH-доступ к удаленному компьютеру, например, с аутентификацией с ключом хоста. Чтобы скопировать ваш открытый ключ на сервер, выполните команду `ssh-copy-id`:

```
duchess@pc:~/.ssh $ ssh-copy-id -i id-server1 duchess@server1
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/id-server1"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are
prompted now it is to install the new keys

Number of key(s) added: 1
Now try logging into the machine, with: "ssh 'duchess@server1'"
and check to make sure that only the key(s) you wanted were added.
```

Попробуйте выполнить вход:

```
duchess@pc:~/.ssh $ ssh -i id-server1 duchess@server1
Enter passphrase for key 'id-server1':
Last login: Sat Jul 11 11:09:53 2021 from 192.168.43.234
Have a lot of fun...
duchess@server1:~$
```

Вы можете применять этот новый ключ для доступа к нескольким удаленным хостам или создать уникальный ключ для каждого из них. Один и тот же ключ для нескольких машин легче использовать, но сложнее заменить. И напротив, если уникальный ключ будет скомпрометирован или утерян, то его потребуется заменить только на одном сервере.

Комментарий

Всегда применяйте парольную фразу для ключей SSH, создаваемых для пользователей-людей, поскольку любой, кто получит доступ к вашим личным ключам, сможет замаскироваться под вас, если вы поленитесь добавить парольную фразу.

Замечательная маленькая утилита `ssh-copy-id` гарантирует копирование открытых ключей в нужное место, то есть в `~/.ssh/authorized_keys` на удаленном хосте, в нужном формате и с нужными разрешениями. Она также гарантирует, что ваш закрытый ключ не будет скопирован по ошибке.

Эта утилита поддерживает следующие параметры:

- `-C` — позволяет добавить комментарий к ключу, который может помочь вспомнить его предназначение;
- `-f` — позволяет задать имя ключа, которое может быть любым. Не забывайте, в каком каталоге находитесь в текущий момент; если не в `~/.ssh`, то укажите путь;
- `-t` — тип ключа: `rsa`, `ecdsa` или `ed25519`;
- `-b` — разрядность; этот параметр поддерживается только для ключей `rsa`. Значение по умолчанию — 2048, максимальная разрядность — 4096. Чем больше разрядность, тем больше вычислительных ресурсов требуется на обработку, но, честно говоря, вы едва ли заметите какую-либо разницу при использовании 4096-разрядного ключа, разве что на старом и слабом оборудовании или на очень загруженных серверах;
- `-i` — сообщает SSH-клиенту, какой ключ должен применяться. Если у вас несколько ключей, то вы должны использовать этот параметр. При наличии нескольких открытых ключей можно столкнуться с сообщением об ошибке Too many authentication failures (Слишком много ошибок аутентификации), если не указать один ключ, поскольку SSH будет пробовать их все, если не указан один конкретный.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- `man 1 ssh`
- `man 1 ssh-keygen`

12.8. Управление несколькими открытыми ключами

Задача

Использовать несколько разных ключей для работы с разными серверами. Как управлять ключами с разными именами?

Решение

При создании новой пары ключей используйте команду `ssh-keygen` с параметром `-f`, чтобы присвоить ключам уникальные имена:

```
duchess@pc:~/.ssh $ ssh-keygen -t rsa -f id-server2
```

После этого можно будет с помощью параметра `-i` выбирать ключ для подключения к конкретному удаленному хосту:

```
duchess@pc:~/.ssh $ ssh -i id-server2 duchess@server2
```

Чтобы упростить управление несколькими открытыми ключами, создайте новый файл `~/.ssh.config`. Он настраивает учетные данные для подключения к различным удаленным хостам; произведя такие настройки, можно выполнять вход с помощью простой короткой команды `ssh foo`. Следующий пример демонстрирует настройки, упрощающие вход на удаленный хост `server2` для пользователя Duchess:

```
Host server2
  HostName server2
  User duchess
    IdentityFile ~/.ssh/id-server2
    IdentitiesOnly yes
```

Теперь Duchess сможет выполнять вход, просто указав значение параметра `Host`:

```
$ ssh server2
```

В этот файл можно добавить все используемые вами открытые ключи, например:

```
Host server3
  HostName server3
  User duchess
    IdentityFile ~/.ssh/id-server3
    IdentitiesOnly yes

Host server3
  HostName server3
  User madmax
    IdentityFile ~/.ssh/id-server3
    IdentitiesOnly yes
```

Комментарий

Ниже кратко описаны параметры из предыдущего примера.

- Строка `Host` определяет начало блока настроек для каждого отдельного хоста. Это метка, которая будет использоваться для входа в систему, и она может быть любой строкой.
- `HostName` — имя хоста удаленного компьютера, полное доменное имя или IP-адрес.
- `User` — имя пользователя на удаленной машине.
- `IdentityFile` — полный путь к файлу с вашим открытым ключом.
- `IdentitiesOnly yes` — указывает `ssh` использовать настройки в `~/.ssh/config` или из параметров командной строки, а не из других источников, если они есть.

Номер порта SSH по умолчанию — 22. Если нужно подключиться к нестандартному порту, например, 2022, то укажите его с помощью параметра `Port`:

```
Port 2022
```

Вы можете присвоить своим ключам какие угодно имена. Я предпочитаю описательные, чтобы они напоминали, для связи с какой машиной предназначен тот или иной ключ.

Не забывайте всегда добавлять парольную фразу в свои закрытые ключи.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- `man 1 ssh_config`
- `man 1 ssh`

12.9. Изменение парольной фразы

Задача

Изменить парольную фразу на выбранном закрытом ключе.

Решение

Используйте команду `ssh-keygen` с параметром `-p`:

```
$ ssh-keygen -p -f ~/.ssh/id-server2
Enter old passphrase:
Key has comment 'backup server2'
Enter new passphrase (empty for no passphrase): passphrase
Enter same passphrase again: passphrase
Your identification has been saved with the new passphrase.
```

Комментарий

Парольные фразы нельзя восстановить. Если вы забудете ее, то вам останется только создать новый ключ с новой парольной фразой.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- `man 1 ssh`
- `man 1 ssh-keygen`

12.10. Автоматическое управление парольными фразами с помощью Keychain

Задача

Хотелось бы использовать какой-то инструмент, запоминающий парольные фразы закрытых ключей.

Решение

Для этой цели создана утилита Keychain. Установите пакет *keychain*, затем скопируйте строки из следующего примера в свой файл `.bashrc`.

Следующий пример демонстрирует, как организовать вход на `server1`, `server2` и `server3`, не вводя каждый раз парольные фразы. Скопируйте эти строки, подставив в них имена своих ключей:

```
keychain ~/.ssh/id-server1 ~/.ssh/id-server2 \
~/.ssh/id-server3 . ~/.keychain/$HOSTNAME-sh
```

Утилита Keychain сохраняет доступность закрытых ключей до завершения сеанса, поэтому вам придется вводить парольные фразы только при запуске системы.

После загрузки графической среды вам может быть не предложено ввести парольные фразы. Попробуйте открыть терминал, и если Keychain и в этом случае не вывела запрос, то войдите в консоль Linux: нажмите комбинацию Ctrl+Alt+F2 и выполните вход в систему. Затем вы должны увидеть примерно такие строки:

```
* keychain 2.8.5 ~ http://www.funtoo.org
 * Found existing ssh-agent: 2016
 * Adding 3 ssh key(s): /home/duchess/.ssh/id-server1
/home/duchess/.ssh/id-server2 /home/duchess/.ssh/id-server3
Enter passphrase for /home/duchess/.ssh/id-server1:
Enter passphrase for /home/duchess/.ssh/id-server2:
Enter passphrase for /home/duchess/.ssh/id-server3:
 * ssh-add: Identities added: /home/duchess/.ssh/id-server1
/home/duchess/.ssh/id-server2 /home/duchess/.ssh/id-server3
```

Комментарий

Начальная точка в команде `. ~/.keychain/$HOSTNAME-sh` — это короткое имя команды `source`, означающей использование указанного файла.

Элемент `$HOSTNAME` сообщает утилите Keychain имя переменной среды, в которой хранится имя хоста. Вы можете сами посмотреть содержимое этой переменной:

```
$ echo $HOSTNAME
pc
```

Keychain — это диспетчер ключей для *ssh-agent* и *gpg-agent*, который кэширует парольные фразы SSH и GPG, пока компьютер включен. Вы можете выйти из системы и снова войти в нее, но вам не придется вводить парольные фразы заново — только после перезапуска системы.

Хорошая альтернатива — утилита *gnome-keyring*, работающая в графической среде. Утилита имеет графический интерфейс для просмотра и управления ключами SSH и GPG и включает диспетчер паролей. В большинстве систем она отображается в меню как Passwords and Keys (Пароли и ключи). Но у нее есть два недостатка: она не подходит для использования в системах без графического интерфейса и не обеспечивает доступность парольных фраз для заданий cron (см. рецепт 12.11).

Дополнительная информация

- Funtoo Keychain (https://oreil.ly/rljaf).

12.11. Использование Keychain для доступа к парольным фразам из заданий cron

Задача

Использовать cron для автоматизации задач, таких как запуск резервного копирования с `rsync` на удаленный хост. Но, что бы вы ни делали, как бы ни исхитрялись, у вас ничего не получается и ваши задания резервного копирования выводят только сообщения об ошибке аутентификации.

Решение

Чтобы настроить управление вашими закрытыми ключами для заданий cron с помощью Keychain, создайте сценарий, который будет запускаться из cron. Например, ниже представлен сценарий, который называется `duchess-backup-server1` и выполняет резервное копирование с `rsync`:

```
#!/bin/bash
source $HOME/.keychain/${HOSTNAME}-sh
/usr/bin/rsync -ae "ssh -i /home/duchess/.ssh/id-server3" /home/duchess/ \
duchess@server1:/backups/
```

С помощью команды `chmod` сделайте сценарий выполняемым:

```
$ chmod +x duchess-backup-server1
```

Добавьте в crontab строку, определяющую расписание запуска сценария, например, каждый день в 22:15:

```
15 22 * * * /home/duchess/duchess-backup-server1
```

Комментарий

В примере сценария строка, начинающаяся с `/usr/bin/rsync`, должна находиться на одной строке.

Планировщик заданий cron работает в своей особой и ограниченной среде, поэтому сценарий должен настроить необходимые переменные среды и использовать утилиту Keychain для доступа к требуемым ключам.

Дополнительная информация

- `man 1 crontab`
- Funtoo Keychain (https://oreil.ly/rljaf).

12.12. Защищенное туннелирование сеанса X через SSH

Задача

Нужна возможность запускать графические приложения на удаленном хосте. Вы знаете, что система X Window имеет встроенную сетевую поддержку, но весь трафик отправляется в открытом виде, что небезопасно, и было бы желательно обезопасить себя.

Решение

Туннелирование X через SSH не требует дополнительного программного обеспечения. Для начала выполните следующие команды, чтобы узнать, какой протокол поддерживает ваша система: X11 или Wayland. В следующих примерах показаны оба результата:

```
$ echo $XDG_SESSION_TYPE
x11
$ echo $XDG_SESSION_TYPE
wayland
$ loginctl show-session "$XDG_SESSION_ID" -p Type
Type=x11
$ loginctl show-session "$XDG_SESSION_ID" -p Type
Type=wayland
```

`loginctl` — это часть systemd.

Если у вас используется Wayland, то вы не сможете организовать туннелирование через SSH, поскольку Wayland не поддерживает работу по сети.

Если ваша система использует X11, то настройте передачу данных по протоколу X11 в `/etc/ssh/sshd_config` на удаленном компьютере:

```
X11Forwarding yes
```

Ниже представлен пример настройки туннелирования X через SSH с использованием параметра `-Y`:

```
duchess@pc:~$ ssh -Yi id-server1 duchess@server1
Last login: Thu Jul 9 09:26:09 2021 from 192.168.43.80
Have a lot of fun..
duchess@server1:~$
```

Теперь вы сможете запускать графические приложения, правда, только по одному за раз, например, игру (рис. 12.1):

```
duchess@server1:~$ kmahjongg
```



Рис. 12.1. Игра KMahjongg, запущенная на удаленном сервере

Комментарий

X-сервер работает со смещением `X11DisplayOffset 10`, указанным в `/etc/ssh/sshd.conf`. Это позволяет избежать конфликтов с существующими X-сеансами. Ваш обычный локальный сеанс X — `:0.0`, а первый удаленный сеанс X — `:10.0`. Вы можете увидеть это собственными глазами, выполнив следующие команды на локальном компьютере. Первая команда определяет принадлежность к локальной командной строке:

```
duchess@pc:~$ echo $DISPLAY
:0.0
```

Второй пример выполнялся в удаленной командной оболочке через SSH:

```
duchess@server1:~ssh $ echo $DISPLAY
localhost:10.0
```

Для работы с удаленной системой достаточно включить ее — не нужны никакие локальные пользователи, чтобы войти в систему, и даже не нужен X-сервер. Он должен работать только на клиентском компьютере.

Дополнительная информация

- man 1 sshd
- man 1 ssh_config

12.13. Открытие сеанса SSH и запуск команды одной строкой

Задача

Вам нужно выполнить единственную команду на удаленном компьютере, и вы подумали, что было бы неплохо запустить ее без входа в систему и последующего выхода из нее. В конце концов, это правда, что лень — добродетель системных администраторов!

Решение

OpenSSH позволяет сделать это. Ниже представлен пример перезапуска Postfix:

```
$ ssh mailadmin@server2.example.com sudo systemctl restart postfix
```

Вам может быть предложено ввести пароль `sudo`, но все равно вы сэкономите целый шаг.

Следующая команда запустит игру GNOME Sudoku, для чего требуется X Window System:

```
$ ssh -Y duchess@laptop /usr/games/gnome-sudoku
```

Комментарий

Другой способ — прибегнуть к аутентификации с открытым ключом для пользователя root, благодаря чему отпадает необходимость вызывать `sudo` (рецепт 12.7).

Дополнительная информация

- `man 1 ssh`

12.14. Монтирование удаленной файловой системы через sshfs

Задача

OpenSSH работает быстро и эффективно, и даже туннелирование X-приложений через OpenSSH не слишком замедляет их. Но теперь вам хотелось бы иметь возможность быстро отредактировать множество удаленных файлов, избегая запуска диспетчера файлов с графическим интерфейсом через SSH.

Решение

В этом вам поможет *sshfs* — инструмент, предназначенный для монтирования удаленных файловых систем и последующего доступа к ним как к локальной файловой системе, без хлопот по настройке сервера NFS или Samba.

Установите пакет sshfs, который также должен установить FUSE, Filesystem in Userspace (файловую систему в пространстве пользователя). Кроме того, вам понадобится локальный каталог с разрешением на запись, который будет играть роль точки монтирования:

```
duchess@pc:~$ mkdir sshfs
```

Ниже представлен пример монтирования удаленного каталога в свой локальный каталог `sshfs`. Это пример монтирует домашний каталог `duchess@server2` в каталог `sshfs` на `duchess@pc`:

```
duchess@pc:~$ sshfs duchess@server2: sshfs/
```

После этого вы сможете обращаться к удаленной файловой системе, как если бы она была локальной:

```
duchess@pc:~$ ls sshfs
Desktop
Documents
Downloads
[...]
```

Файлы в такой смонтированной удаленной файловой системе будут доступны и из командной строки, и из диспетчера файлов с графическим интерфейсом, в точности как локальные файлы.

Приглашение в командной строке не изменится.

Завершив работу с удаленной файловой системой, размонтируйте ее:

```
duchess@pc:~$ fusermount -u sshfs/
```

Команда выше монтирует домашний каталог пользователя Duchess целиком. Однако при желании можно смонтировать только выбранный подкаталог:

```
duchess@pc:~$ sshfs duchess@server2:/home/duchess/arias sshfs/
```

Здесь нельзя использовать символ тильды (~) для краткого обозначения пути к домашнему каталогу `/home/user`, поскольку sshfs не поддерживает его.

Если сетевое соединение ненадежно, подскажите утилите sshfs, что она должна автоматически восстанавливать подключение в случае разрыва:

```
duchess@pc:~$ sshfs duchess@server2:/home/duchess/arias sshfs/ -o reconnect
```

Комментарий

Пользователи, плохо знакомые с sshfs, всегда задают следующие вопросы: «Почему просто не запустить X через SSH?» — или: «Почему просто не задействовать NFS?» Ответ прост: прием с sshfs работает быстрее, чем запуск X через SSH, проще в настройке, чем NFS, и не запрещает использовать NFS, Samba и все, что вам будет угодно.

Дополнительная информация

- `man 1 sshfs`

12.15. Настройка приглашения к вводу в Bash при работе через SSH

Задача

Вы уже знаете, что приглашение к вводу меняется после входа в систему через SSH, и в приглашении начинает отображаться имя удаленного хоста. Но это всего лишь простая подсказка, которую можно по ошибке прочитать неправильно, поэтому хотелось бы иметь особое приглашение к вводу, четко указывающее, что вы работаете с удаленной системой через SSH.

Решение

Настройте приглашение к вводу в Bash на удаленных машинах. Ниже представлен пример изменения цвета приглашения на фиолетовый и добавления слова `ssh`.

Скопируйте эти строки в файл `.bashrc` в удаленной учетной записи, в которую вы хотите войти:

```
if [ -n "$SSH_CLIENT" ]; then text=" ssh"
fi
export PS1='\[\e[0;36m\]\u@\h:\w${text}$\[\e[0m\] '
```

В этом случае, после входа на удаленную машину, приглашение к вводу будет выглядеть так, как показано на рис. 12.2.

```
duchess@pc:~/.ssh$ ssh -i id-server2 duchess@server2
Enter passphrase for key 'id-server2':
Last login: Sat Jul 11 11:09:53 2020 from 192.168.43.234
Have a lot of fun...
duchess@server2:~ssh$
```

Рис. 12.2. Настроенное приглашение к вводу при работе через SSH

В фиолетовый цвет будет окрашено только само приглашение к вводу, а весь остальной текст будет отображаться как обычно.

Комментарий

Настройка приглашения в Bash — это обширная тема, достойная отдельной книги. Пример в этом рецепте можно поправить и подогнать под свой вкус. Совсем не обязательно использовать слово `ssh` или называть переменную `text`; сам текст и имя переменной могут быть какими угодно. Например, вы можете выводить приглашение с текстом «суперзашифрованный сеанс», а переменную назвать `sekkret-squirl`.

`[\ e [0; 31m \]` — это блок, определяющий цвет текста. Чтобы изменить цвет, достаточно лишь изменить числа в данном блоке.

`[\ e [0m \]` отключает пользовательские цвета, возвращая цвет вывода к нормальным настройкам оболочки. Ниже представлены некоторые коды цветов:

- черный — 0; 30;
- синий — 0; 34;
- зеленый — 0; 32;
- голубой — 0; 36;
- красный — 0; 31;
- фиолетовый — 0; 35;
- коричневый — 0; 33;
- светло-серый — 0; 37;
- темно-серый — 1; 30;
- светло-синий — 1; 34;
- светло-зеленый — 1; 32;
- светло-голубой — 1; 36;
- светло-красный — 1; 31;
- светло-фиолетовый — 1; 35;
- желтый — 1; 33;
- белый — 1; 37.

Данная настройка основана на проверке наличия переменной среды `SSH_CLIENT`, которая присутствует только при наличии активного SSH-соединения. Вы можете сами убедиться в этом на удаленном хосте:

```
$ echo $SSH_CLIENT
192.168.43.234 51414 22
```

Так Bash сможет узнать, что нужно использовать нестандартное приглашение к вводу. Если запустить эту команду на машине, где нет активных сеансов SSH, то она вернет пустую строку.

Дополнительная информация

- `man 1 bash`
- Bash Prompt HOWTO, Chapter 6 (https://oreil.ly/QXWmT).

12.16. Список поддерживаемых алгоритмов шифрования

Задача

Чтобы соблюдать согласованные правила, вам нужно знать, какие алгоритмы шифрования поддерживает OpenSSH.

Решение

В OpenSSH есть команда, с помощью которой можно попросить перечислить все поддерживаемые алгоритмы: `ssh -Q` *<параметры_запроса>*. Список алгоритмов можно вывести, передав параметр запроса `help`:

```
$ ssh -Q help
cipher
cipher-auth
compression
kex
kex-gss
key
key-cert
key-plain
key-sig
mac
protocol-version
sig
```

Следующий пример использует параметр `sig` для перечисления сигнатур алгоритмов:

```
$ ssh -Q sig
ssh-ed25519
sk-ssh-ed25519@openssh.com
ssh-rsa
rsa-sha2-256
rsa-sha2-512
ssh-dss
ecdsa-sha2-nistp256
ecdsa-sha2-nistp384
ecdsa-sha2-nistp521
sk-ecdsa-sha2-nistp256@openssh.com
```

Комментарий

Ниже представлено краткое описание каждого параметра:

- `cipher` перечисляет поддерживаемые симметричные алгоритмы шифрования;
- `cipher-auth` перечисляет поддерживаемые симметричные алгоритмы шифрования, которые также поддерживают шифрование аутентификации;
- `compression` перечисляет поддерживаемые типы сжатия;
- `mac` перечисляет поддерживаемые коды целостности сообщений, помогающие проверить целостность и аутентичность данных в сообщении;
- `kex` перечисляет алгоритмы обмена ключами;
- `kex-gss` перечисляет алгоритмы обмена ключами GSSAPI (Generic Security Service Application Program Interface — прикладной программный интерфейс для служб безопасности);
- `key` перечисляет типы ключей;
- `key-cert` перечисляет типы ключей-сертификатов;
- `key-plain` перечисляет типы ключей-несертификатов;
- `key-sig` перечисляет все типы ключей и сигнатуры алгоритмов;
- `protocol-version` перечисляет поддерживаемые версии протокола SSH (на момент написания этих строк поддерживалась только одна версия — 2);
- `sig` перечисляет сигнатуры алгоритмов.

Дополнительная информация

- Сайт OpenSSH (https://openssh.com).
- Книга Serious Cryptography Жана-Филиппа Аумассона (Jean-Philippe Aumasson) (No Starch Press).

ГЛАВА 13

Безопасный удаленный доступ с OpenVPN

Открытая виртуальная частная сеть (Open Virtual Private Network, OpenVPN) создает соединение, шифруемое с использованием протокола TLS/SSL, между разными сетями в физически разных местах, например между филиалом и главным офисом или компьютером сотрудника, работающего удаленно, и сетью компании. Такое соединение называется шифрованным *туннелем*, безопасным транспортом, защищающим соединение от большого Интернета. Технология OpenVPN основана на OpenSSL, поэтому знание OpenSSL будет нелишним.



Если вы уже знакомы с OpenVPN, то можете сразу перейти к рецептам 13.5, 13.6 и 13.7, чтобы посмотреть, как создавать сертификаты шифрования и настраивать клиентов и серверы. Если вы новичок в VPN, то опробуйте каждый рецепт один за другим. Не торопитесь; виртуальные частные сети сложны и привередливы. Тщательно протестируйте свои настройки перед развертыванием в производственных условиях.

Обзор OpenVPN

VPN — это безопасное расширение сети, которое делает доступными для удаленных сотрудников все службы, имеющиеся в распоряжении локальных пользователей, поэтому удаленные пользователи обладают теми же возможностями, что и пользователи, физически присутствующие на работе. Они могут обращаться к вашим локальным веб-серверам, электронной почте, общим папкам, чат-серверам, приложениям для видеоконференций, внутренним вики-страницам — ко всему, что вы оградили от внешнего мира и что доступно только пользователям внутри вашей сети. VPN не похожа на SSH, которая соединяет отдельные компьютеры. VPN связывает сети и отдельные хосты с сетями.

В этой главе вы узнаете, как настроить сервер и клиентов OpenVPN и как создать и управлять инфраструктурой открытых ключей (Public Key Infrastructure, PKI) для аутентификации и шифрования. Ваш сервер будет аутентифицировать и защищать любых клиентов: Linux, macOS, Windows, Android и iOS.

OpenVPN — проект с открытым исходным кодом, с бесплатными и коммерческими вариантами. Бесплатные сервер и клиент составляют пакет *openvpn*, доступный во всех дистрибутивах Linux и на сайте OpenVPN Community Downloads (https://oreil.ly/vwEAs). Коммерческие варианты включают OpenVPN Access Server — промышленный сервер с дополнительными инструментами управления и поддержкой облачных вычислений. Предлагаемые индивидуальные тарифные планы требуют только установки клиента и предоставляют доступ к глобальной сети серверов OpenVPN.

Настоящая VPN надежна, поскольку никому не доверяет и требует аутентификации конечных точек, когда сервер и клиент аутентифицируются друг у друга. Большинство коммерческих сетей TLS/SSL VPN этого не делают, а доверяют всем клиентам, как, например, торговые сайты. Это более гибкий подход, позволяющий пользователям входить в систему из любого места с любого устройства. Это очень удобно, когда не требуется устанавливать и настраивать клиентское программное обеспечение и копировать ключи шифрования. Но для внутренней сети, предназначенной не для всех, последнее, что вам нужно, — пользователи, которые входят в систему со случайных компьютеров или смартфонов, зараженных клавиатурными шпионами и вредоносными программами, и встречают тут теплый прием.

Центр сертификации

Центр сертификации (Certificate Authority, CA) — самая важная часть работы сервера OpenVPN. ЦС создает цифровые сертификаты и удостоверяет право владения открытыми ключами. Щелкните на значке с изображением замка в браузере, чтобы увидеть общедоступный сертификат сайта и узнать, какой центр сертификации его подписал. ЦС — доверенный орган, и именно поэтому так много сайтов используют коммерческие центры сертификации. Внутри своей организации вы можете использовать самозаверенные (или самоподписанные) сертификаты, подобные тем, которые мы будем создавать в данной главе. Но сайты, ориентированные на клиентов, должны применять сертификаты, выданные коммерческими центрами сертификации. Использование услуг ЦС избавляет от хлопот с хранением копий клиентских сертификатов на вашем сервере OpenVPN; серверу достаточно знать, что сертификат клиента аутентифицирован вашим ЦС.

SSL и TLS

Secure Sockets Layer (SSL) и Transport Layer Security (TLS) — это криптографические протоколы. TLS произошел от SSL. В настоящее время все версии SSL считаются устаревшими, так же как TLS 1.0 и TLS 1.1. Используйте TLS 1.2 или 1.3 и отключите все остальные (см. рецепт 13.10). Старые версии не могут обеспечить достаточный уровень безопасности, поэтому не позволяйте никому уговорить вас использовать их.

TUN/TAP

Устройства *TUN* и *TAP* — это виртуальные сетевые интерфейсы. Они встроены в ядро Linux, и вам ничего не нужно делать, чтобы сделать их доступными. Устройство TUN предназначено для маршрутизируемых сетей, а устройство TAP — для мостовых сетей. В конфигурационных файлах вашего сервера и клиента указывается, какое из этих устройств использовать.



Надежная безопасность требует усилий

Поддержание безопасности на высоком уровне требует постоянного расширения и углубления знаний и умений. Цель этой главы — показать, как настроить надежную VPN, достаточно удобную для пользователя. Однако существует множество дополнительных методов повышения надежности VPN, таких как использование клиентских сертификатов с коротким сроком службы, расширенная аутентификация, аппаратные устройства, SELinux, chroot-окружения, короткие тайм-ауты паролей и многое другое. Если вам требуется сверхвысокая безопасность, то проконсультируйтесь с опытными профессионалами.

13.1. Установка OpenVPN, сервера и клиента

Задача

Научиться устанавливать openVPN.

Решение

На сайте проекта OpenVPN (https://openvpn.net) доступны для скачивания бесплатная версия OpenVPN, распространяемая с открытым исходным кодом, и коммерческая версия OpenVPN Access Server. В этой главе рассказывается о бесплатной версии OpenVPN.

Установите в своей системе Linux пакет *openvpn*. (Как обычно, имя пакета в вашем конкретном дистрибутиве Linux может немного отличаться.) Выбирайте самую последнюю версию из доступных, но не ниже 2.4.5. Пакет содержит и сервер, и клиент. Архивы с исходным кодом и установочные пакеты для Windows доступны в разделе скачивания общедоступных версий на сайте OpenVPN (https://oreil.ly/vwEAs).

Своим клиентам можете попробовать установить бесплатную версию клиента OpenVPN Connect (https://oreil.ly/vQugl), доступную для Linux, macOS, Android, iOS и Windows. Она разработана для использования с коммерческим сервером OpenVPN Access Server, но поддерживает также работу с бесплатным сервером OpenVPN.

Бесплатную версию клиента OpenVPN для Android можно найти и в магазине Google Play.

См. рецепт 13.9, чтобы узнать, как с помощью файлов в формате `.ovpn` упростить настройку клиента.

Комментарий

В Linux поддержка OpenVPN должна быть установлена и на сервере OpenVPN, и на всех клиентах. Пакет OpenVPN включает поддержку обеих сторон, клиента и сервера.

Ubuntu, Fedora и openSUSE включают дополнительные пакеты, обеспечивающие интеграцию с NetworkManager, что упрощает управление, подключение и отключение от VPN.

NetworkManager-openvpn (Fedora, openSUSE) и *network-manager-openvpn* (Ubuntu) интегрируют OpenVPN с NetworkManager. Если вы используете среду GNOME (такую как GNOME, Xfce, Cinnamon или Mate), то вам также понадобится *NetworkManager-openvpn-gnome* (openSUSE, Fedora) или *network-manager-openvpn* (Ubuntu).

Сервер OpenVPN Access Server можно скачать бесплатно и использовать для подключения до двух клиентов одновременно без покупки лицензии. Он поддерживает дополнительные возможности, такие как веб-интерфейс администрирования и автоматические настройки с помощью бесплатного клиента OpenVPN Connect. Если вы начнете с бесплатной версии OpenVPN, а затем решите перейти на OpenVPN Access Server, то все, что вы узнали, используя бесплатную версию OpenVPN, в равной степени будет применимо и при использовании Access Server.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.2. Настройка простого подключения для тестирования

Задача

Вы решили настроить простое подключение OpenVPN для тестирования, чтобы получить представление об особенностях работы и заодно проверить возможность подключения.

Решение

Следующий простой тест создает незашифрованный туннель между двумя компьютерами Linux, находящимися в одной сети. На обоих должна быть установлена поддержка OpenVPN. Сначала убедитесь, что демон OpenVPN не запущен ни на одном из хостов, и если это не так, то остановите его:

```
$ systemctl status openvpn@.openvpn1.service
● openvpn.service - OpenVPN service
     Loaded: loaded (/lib/systemd/system/openvpn.service; enabled; vendor prese>
     Active: active (exited) since Sun 2021-01-10 13:43:18 PST; 33min ago
[...]
$ sudo systemctl stop openvpn@.openvpn1.service
```



Используйте отдельную подсеть для своей VPN

Используйте отдельную подсеть для своего туннеля OpenVPN; например, host1 и host2 находятся в сети 192.168.43.0/24, поэтому в примере для создания туннеля VPN задействуется частное адресное пространство 10.0.0.0/24.

Далее в примерах используются два компьютера с именами `host1` и `host2`. В первом примере создается VPN-туннель от `host1` к `host2`:

```
[madmax@host1 ~]$ sudo openvpn  --remote host2 --dev tun0 --ifconfig 10.0.0.1 \
10.0.0.2
Sat Jan  9 14:40:34 2021 disabling NCP mode (--ncp-disable) because not in P2MP
  client or server mode
```

```
Sat Jan  9 14:40:34 2021 OpenVPN 2.4.8 x86_64-redhat-linux-gnu [SSL (OpenSSL)]
  [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Jan 29 2020
Sat Jan  9 14:40:34 2021 library versions: OpenSSL 1.1.1d FIPS  10 Sep 2019,
  LZO 2.10
Sat Jan  9 14:40:34 2021 ******* WARNING *******: All encryption and
  authentication features disabled -- All data will be tunnelled as clear text
  and will not be protected against man-in-the-middle changes. PLEASE DO
  RECONSIDER THIS CONFIGURATION!
Sat Jan  9 14:40:34 2021 TUN/TAP device tun0 opened
Sat Jan  9 14:40:34 2021 /sbin/ip link set dev tun0 up mtu 1500
Sat Jan  9 14:40:34 2021 /sbin/ip addr add dev tun0 local 10.0.0.1 peer 10.0.0.2
Sat Jan  9 14:40:34 2021 TCP/UDP: Preserving recently used remote address:
  [AF_INET]192.168.122.239:1194
Sat Jan  9 14:40:34 2021 UDP link local (bound): [AF_INET][undef]:1194
Sat Jan  9 14:40:34 2021 UDP link remote: [AF_INET]192.168.122.239:1194
```

А в этом примере — туннель от `host2` к `host1`:

```
[stash@host2 ~]$ sudo openvpn --remote host1 --dev tun0 --ifconfig 10.0.0.2 \
10.0.0.1
Sat Jan  9 14:50:53 2021 disabling NCP mode (--ncp-disable) because not in P2MP
  client or server mode
Sat Jan  9 14:50:53 2021 OpenVPN 2.4.7 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO]
  [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Sep  5 2019
Sat Jan  9 14:50:53 2021 library versions: OpenSSL 1.1.1f  31 Mar 2020, LZO 2.10
Sat Jan  9 14:50:53 2021 ******* WARNING *******: All encryption and
  authentication features disabled -- All data will be tunnelled as clear text
  and will not be protected against man-in-the-middle changes. PLEASE DO
  RECONSIDER THIS CONFIGURATION!
Sat Jan  9 14:50:53 2021 TUN/TAP device tun0 opened
Sat Jan  9 14:50:53 2021 /sbin/ip link set dev tun0 up mtu 1500
Sat Jan  9 14:50:53 2021 /sbin/ip addr add dev tun0 local 10.0.0.2 peer 10.0.0.1
Sat Jan  9 14:50:53 2021 TCP/UDP: Preserving recently used remote address:
  [AF_INET]192.168.122.52:1194
Sat Jan  9 14:50:53 2021 UDP link local (bound): [AF_INET][undef]:1194
Sat Jan  9 14:50:53 2021 UDP link remote: [AF_INET]192.168.122.52:1194
Sat Jan  9 14:51:03 2021 Peer Connection Initiated with
  [AF_INET]192.168.122.52:1194
Sat Jan  9 14:51:04 2021 WARNING: this configuration may cache passwords in
  memory -- use the auth-nocache option to prevent this
Sat Jan  9 14:51:04 2021 Initialization Sequence Completed
```

Соединение считается успешно установленным, когда на обоих хостах появится сообщение Initialization Sequence Completed (процедура инициализации завершена). Проверьте свои соединения, выполнив эхо-запрос через интерфейс `tun0` на обоих хостах:

```
[madmax@host1 ~]$ ping -I tun0 10.0.0.2
PING 10.0.0.2 (10.0.0.2) from 10.0.0.1 tun0: 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.515 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.436 ms
```

```
[stash@host2 ~]$ ping -I tun0 10.0.0.1
PING 10.0.0.1 (10.0.0.1) from 10.0.0.2 tun0: 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.592 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.534 ms
```

Нажмите комбинацию Ctrl+C на каждом хосте, чтобы остановить отправку эхо-запросов, и затем еще раз, чтобы закрыть туннели.

Комментарий

Этот простой тест показывает, как работает OpenVPN. Он создает виртуальный сетевой интерфейс `tun0` на обоих хостах, а затем направляет сетевой трафик через него. Тест не создает шифрованного соединения, о чем предупреждает сообщение ******* WARNING *******: All encryption and authentication features disabled (ВНИМАНИЕ: все функции шифрования и аутентификации выключены) в выводе команды.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- systemd.unit (https://oreil.ly/2AAEe).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.3. Настройка простого шифрования со статическими ключами

Задача

Добавить шифрование для OpenVPN максимально простым способом.

Решение

Самый простой способ настроить шифрование — применить общие статические ключи. Они удобны для тестирования, но не подходят для использования в промышленном окружении. (См. ниже подраздел «Комментарий», чтобы узнать об их недостатках.) В этом рецепте вы узнаете, как создавать и использовать общие

статические ключи, а также как создавать простые файлы конфигурации для сервера и клиента.

Выполните эти шаги:

1) создайте общий статический ключ и передайте на два хоста;
2) создайте конфигурационные файлы для сервера и клиента;
3) запустите OpenVPN на обоих хостах, использовав ссылки на их конфигурационные файлы.

В следующих примерах сервер OpenVPN находится на `server1`, клиент — на `client1`, а новый ключ — в файле `myvpn.key`. Но, вообще говоря, ключи можно хранить в файлах с любыми именами.

Создайте новый каталог на сервере OpenVPN для хранения ключей, затем создайте новый статический ключ:

```
$ sudo mkdir /etc/openvpn/keys
$ sudo openvpn --genkey --secret myvpn.key
```

Скопируйте ключ на клиентскую машину:

```
$ scp myvpn.key client1:/etc/openvpn/keys/
Password:
myvpn.key                       100%  636   142.7KB/s   00:00
```

Создайте конфигурационный файл с настройками сервера. Пример для этого рецепта находится в файле `/etc/openvpn/server1.conf`, но вы можете назвать его как угодно. Используйте отдельную подсеть для вашего туннеля OpenVPN; например, `server1` и `client1` находятся в подсети 192.168.43.0/24, поэтому в примере для VPN-туннеля используется частное адресное пространство 10.0.0.0/24. Tun-адрес сервера — 10.0.0.1:

```
# server1.conf
dev tun
ifconfig 10.0.0.1 10.0.0.2
secret /etc/openvpn/keys/myvpn.key
local 192.168.43.184
```

`local` — это IP-адрес сервера в локальной сети.

Создайте конфигурационный файл с настройками клиента. Tun-адрес клиента — 10.0.0.2:

```
# client1.conf
dev tun
ifconfig 10.0.0.2 10.0.0.1
secret /etc/openvpn/keys/myvpn.key
remote 192.168.43.184
```

Убедитесь, что демон OpenVPN не запущен ни на сервере, ни на клиенте:

```
$ sudo systemctl stop openvpn
```

Запустите OpenVPN на сервере и клиенте:

```
[server1 ~] $ sudo openvpn /etc/openvpn/server1.conf
```

```
[client1 ~] $ sudo openvpn /etc/openvpn/client1.conf
```

Когда на обоих хостах появится сообщение Initialization Sequence Completed (Процедура инициализации завершена), у вас будет установленное соединение. Пошлите эхо-запросы каждому хосту через интерфейс виртуальной частной сети `tun`:

```
[server1 ~] $ ping -I tun0 10.0.0.1
[client1 ~] $ ping -I tun0 10.0.0.2
```

Нажмите комбинацию Ctrl+C на обоих хостах, чтобы закрыть соединение.

Комментарий

Если в выводе команды вы увидели сообщение WARNING: INSECURE cipher with block size less than 128 bit (64 bit). This allows attacks like SWEET32. Mitigate by using a cipher with a larger block size (e.g. AES-256-CBC) (ВНИМАНИЕ: НЕБЕЗОПАСНЫЙ шифр с размером блока менее 128 бит (64 бита). Такой шифр уязвим для атак, подобных SWEET32. Для смягчения последствий используйте шифр с большим размером блока (например, AES-256-CBC)), то добавьте следующую строку в конфигурационные файлы сервера и клиента:

```
cipher AES-256-CBC
```

Самая большая проблема статических ключей заключается в потере совершенной прямой секретности из-за несменяемости статического ключа. Если злоумышленник найдет способ перехватить ваш сетевой трафик, а затем перехватить и взломать ваш ключ шифрования, то сможет расшифровать все, что перехватил в прошлом и еще перехватит в будущем. Инфраструктура открытых ключей OpenVPN PKI использует сложный процесс создания ключей сеансов, которые нигде не хранятся и регулярно меняются. Поэтому, в лучшем случае, удачливый злоумышленник сможет расшифровать трафик только одного сеанса, а затем ему придется начать все заново.

Еще один недостаток — необходимость иметь отдельный ключ для каждого клиента и хранить копии всех клиентских ключей на сервере. Инфраструктура открытых ключей PKI делает управление несколькими клиентами менее трудоемким.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- systemd.unit (https://oreil.ly/2AAEe).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.4. Установка EasyRSA для управления инфраструктурой PKI

Задача

Вы решили использовать EasyRSA для создания инфраструктуры открытых ключей (PKI) и управления ею, и вам требуется правильно выполнить установку и настройку.

Решение

Инфраструктура открытых ключей (PKI) может располагаться где угодно, причем не обязательно на сервере OpenVPN. Вы должны создать сертификаты сервера и клиента в PKI, а затем скопировать их на соответствующие хосты.

Вы можете установить пакет *easy-rsa* или получить самую свежую версию на странице проекта EasyRSA Releases (https://oreil.ly/LtAKu) в GitHub.

Fedora и Ubuntu помещают все файлы EasyRSA в рабочий каталог `/usr/share/`. Он не является самым лучшим, поскольку перезаписывается системными обновлениями. Создайте новый каталог, которым управляете только вы, и в месте, где не нужны права root, как в нашем примере с пользователем Duchess, который создал каталог `/home/duchess/mypki`:

```
~$ mkdir mypki
```

В Fedora и Ubuntu скопируйте каталог `/usr/share/easy-rsa` в созданный вами каталог:

```
~$ sudo cp -r /usr/share/easy-rsa mypki
```

В результате будет создан каталог `mypki/easy-rsa`. Проверьте его разрешения; владельцами всего, что находится в этом каталоге, должны быть вы и ваша группа.

openSUSE выполняет установку правильно и помещает файлы конфигурации в `/etc/easy-rsa`, команду `easyrsa` — в `/usr/bin`, а файлы документации и лицензий — в `/usr/share/`. В этом случае вам не нужно ничего перемещать или беспокоиться о разрешениях.

Комментарий

Для создания инфраструктуры PKI и управления ею не нужны права root. Вы можете разместить ее где угодно, и она должна быть отделена от вашей конфигурации OpenVPN, то есть находиться в отдельном каталоге или на отдельном компьютере. Те, кто давно работает с OpenVPN, рекомендуют устанавливать инфраструктуру на хорошо защищенную машину, не подключенную к Интернету.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- systemd.unit (https://oreil.ly/2AAEe).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.5. Создание инфраструктуры PKI

Задача

После установки EasyRSA (см. рецепт 13.4) нужно правильно настроить инфраструктуру открытых ключей (Public Key Infrastructure, PKI).

Решение

Правильно настроенная инфраструктура PKI необходима для безопасной работы сервера OpenVPN. В этом рецепте мы создадим ее с помощью EasyRSA,

что намного проще процесса с использованием команды `openssl`. Создание инфраструктуры PKI включает следующие шаги:

1) создать собственный сертификат центра сертификации (CA) для подписи сертификатов сервера и клиента. Он должен храниться отдельно от конфигурации сервера OpenVPN — в отдельном каталоге или на отдельном компьютере;
2) создать и подписать сертификат сервера OpenVPN;
3) создать и подписать клиентские сертификаты;
4) скопировать сертификаты сервера и клиентов в `/etc/openvpn/keys` на соответствующие компьютеры. (Имя каталога может быть другим, отличным от `keys`.)

В следующих примерах все команды запускаются из каталога `/home/duchess/mypki/`.

Перейдите в каталог инфраструктуры и выполните команду ее инициализации:

```
~$ cd mypki
~/mypki $ easyrsa init-pki

init-pki complete; you may now create a CA or requests.
Your newly created PKI dir is: /home/duchess/mypki/pki
```

Она создаст пустую структуру каталогов для новой PKI. Затем создайте новый центр сертификации. ЦС создает и подписывает сертификаты сервера и клиентов. Защитите его надежной парольной фразой и задайте желаемое общее имя (`Common Name`) для вашего центра сертификации:

```
~/mypki $ easyrsa build-ca
[...]
Enter New CA Key Passphrase:passphrase
Re-Enter New CA Key Passphrase:passphrase
[...]
Common Name (eg: your user, host, or server name) [Easy-RSA CA]: vpnserver1
[...]
CA creation complete and you may now import and sign cert requests.
Your new CA certificate file for publishing is at:
/home/duchess/mypki/pki/ca.crt
```



Если вы увидите сообщение RAND_load_file: Cannot open file:crypto/rand/randfile.c:98:Filename=/mypki/pki/.rnd (RAND_load_file: Невозможно открыть файл: crypto/rand/randfile.c:98:Filename=/mypki/pki/.rnd), то просто проигнорируйте его, поскольку оно не несет никакого смысла. Вы можете предотвратить появление этого сообщения, отыскав файл openssl-easy-rsa.cnf и закомментировав строку RANDFILE в начале.

Сгенерируйте запрос на создание пары ключей и подписание сертификата для вашего сервера OpenVPN. Обычно не принято добавлять парольную фразу в закрытый ключ сервера, но вы можете защитить его паролем, если хотите, опустив параметр `nopass`. Парольная фраза обеспечивает надежную защиту, но тогда вам придется вводить ее каждый раз при перезапуске сервера:

```
~/mypki $ easyrsa gen-req vpnserver1 nopass

Using SSL: openssl OpenSSL 1.1.1d 10 Sep 2019
Generating a RSA private key
.............................+++++
...............................................................++++++
writing new private key to '/home/duchess/mypki/pki/private/vpnserver1.key.NYjr5y
c9kj'
[...]
Common Name (eg: your user, host, or server name) [vpnserver1]:

Keypair and certificate request completed. Your files are:
req: /home/duchess/mypki/pki/reqs/vpnserver1.req
key: /home/duchess/mypki/pki/private/vpnserver1.key
```

Сгенерируйте запрос на создание пары ключей и подписание сертификата для клиента. Закрытые ключи клиентов должны защищаться паролем, особенно это относится к мобильным клиентам:

```
~/mypki $ easyrsa gen-req vpnclient1

Using SSL: openssl OpenSSL 1.1.1d 10 Sep 2019
Generating a RSA private key
................+++++
...................................................................+++++
writing new private key to '/home/duchess/mypki/pki/private/vpnclient1.key.bicpOc
EC5S'
Enter PEM pass phrase:passphrase
Verifying - Enter PEM pass phrase:passphrase
[...]
Common Name (eg: your user, host, or server name) [vpnclient1]:

Keypair and certificate request completed. Your files are:
req: /home/duchess/mypki/pki/reqs/vpnclient1.req
key: /home/duchess/mypki/pki/private/vpnclient1.key
```

Подпишите запросы, используя общие имена (`Common Name`) сервера и клиента. Используйте только имена; если вы введете их пути, то это приведет к ошибке:

```
~/mypki $ easyrsa sign-req server vpnserver1
Using SSL: openssl OpenSSL 1.1.1d 10 Sep 2019
You are about to sign the following certificate.
Please check over the details shown below for accuracy. Note that this request
has not been cryptographically verified. Please be sure it came from a trusted
```

```
source or that you have verified the request checksum with the sender.

Request subject, to be signed as a server certificate for 1080 days:

subject=
    commonName                = vpnserver1

Type the word 'yes' to continue, or any other input to abort.
  Confirm request details: yes
Using configuration from /home/duchess/mypki/pki/safessl-easyrsa.cnf
Enter pass phrase for /home/duchess/mypki/pki/private/ca.key:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
commonName            :ASN.1 12:'vpnserver1'
Certificate is to be certified until Jan 27 20:09:12 2024 GMT (1080 days)

Write out database with 1 new entries
Data Base Updated

Certificate created at: /home/duchess/mypki/pki/issued/vpnserver1.crt

mypki $ easyrsa sign-req client vpnclient1
[...]
Certificate created at: /home/duchess/mypki/pki/issued/vpnclient1.crt
```

Сгенерируйте параметры протокола Диффи — Хеллмана (Diffie — Hellman) для сервера; это займет минуту или две. Данная команда должна запускаться на сервере OpenVPN:

```
$ easyrsa gen-dh
Using SSL: openssl OpenSSL 1.1.1d 10 Sep 2019
Generating DH parameters, 2048 bit long safe prime, generator 2
This is going to take a long time
...........................................+...........
..........+...........................................
[...]
DH parameters of size 2048 created at /home/duchess/mypki/pki/dh.pem
```

Создайте на сервере ключ HMAC (Hash-based Message Authentication Code — код аутентификации сообщения на основе хеш-функции):

```
$ openvpn --genkey --secret ta.key
```

Скопируйте `vpnclient1.key`, `vpnclient1.crt`, `ca.crt` и `ta.key` в каталог `/etc/openvpn/keys` на машине `client1`.

Скопируйте `vpnserver1.key`, `vpnserver1.crt`, `ca.crt`, `dh.pem` и `ta.key` в каталог `/etc/openvpn/keys` на машине `server1`.

После подписания запроса на подпись сертификата можно удалить все файлы `*.req`.

Таблица 13.1 напомнит вам, какие файлы где должны находиться.

Таблица 13.1. Местоположение ключей сервера и клиента

Имя	Местоположение	Открытый	Закрытый
ca.crt	Сервер и клиенты	X	
ca.key	Компьютер с PKI		X
ta.key	Сервер и клиенты		X
dh.pem	Сервер	X	
server.crt	Сервер	X	
server.key	Сервер		X
client1.crt	Клиент 1	X	
client1.key	Клиент 1		X
client2.crt	Клиент 2	X	
client2.key	Клиент 2		X

Комментарий

Что это за штука — протокол Диффи — Хеллмана? Это механизм шифрования, который позволяет двум хостам создать секретный ключ и совместно использовать его. После аутентификации друг у друга клиент и сервер OpenVPN генерируют дополнительные ключи отправки и получения для шифрования сеанса.

HMAC вычисляет код аутентификации сообщения и проверяет целостность и подлинность сообщения.

Команда `easyrsa init-pki` создает новую инфраструктуру PKI. Команду также можно запустить, чтобы полностью удалить и перестроить существующую инфраструктуру PKI.

Инфраструктуру PKI можно настроить в любом месте, и опытные специалисты по OpenVPN рекомендуют размещать ее на компьютере, не подключенном к Интернету и хорошо защищенном от всех, кто не должен вмешиваться в работу PKI. Если злоумышленник сможет скомпрометировать ваш центр сертификации, то легко сможет проникнуть в вашу сеть. Очевидно, что у вас должен быть

безопасный способ распространения этих файлов: USB-накопитель, команда `scp`, зашифрованный архив, скачанный с защищенного сервера или отправленный по электронной почте вашим пользователям.

Загляните в каталог PKI, чтобы увидеть, как организованы элементы инфраструктуры.

```
~/mypki $ ls */*
pki/ca.crt               pki/index.txt            pki/index.txt.old
pki/serial               pki/dh.pem               pki/index.txt.attr
pki/openssl-easyrsa.cnf  pki/serial.old           pki/extensions.temp
pki/index.txt.attr.old   pki/safessl-easyrsa.cnf  pki/ta.key

pki/certs_by_serial:
4954C26DB44106B20F1B9DA17CE515E5.pem  DA68CBE53E30923C9BCC3B9F1C5C9011.pem

pki/issued:
vpnclient1.crt  vpnserver1.crt

pki/private:
ca.key  vpnclient1.key  vpnserver1.key

pki/renewed:
certs_by_serial  private_by_serial  reqs_by_serial

pki/reqs:
vpnclient1.req  vpnserver1.req

pki/revoked:
certs_by_serial  private_by_serial  reqs_by_serial
```

Запросы на подпись хранятся в файлах с расширением `.req`, открытые ключи — `.crt` и закрытые ключи — `.key`. Ключи всегда создаются парами, состоящими из открытого и закрытого ключей.



Открытые ключи — для шифрования, закрытые ключи — для расшифровывания

Открытые ключи предназначены для шифрования, а закрытые — для расшифровывания. Закрытые ключи должны храниться в секрете и никому не передаваться. Открытые ключи, напротив, предназначены для передачи другим.

Нажмите на любом подписанном сертификате в диспетчере файлов, чтобы получить информацию о сертификате, как показано на рис. 13.1. Здесь вы увидите много интересного: название центра сертификации, подписавшего его, дату истечения срока действия, серийный номер, отпечаток, подпись и многое другое.

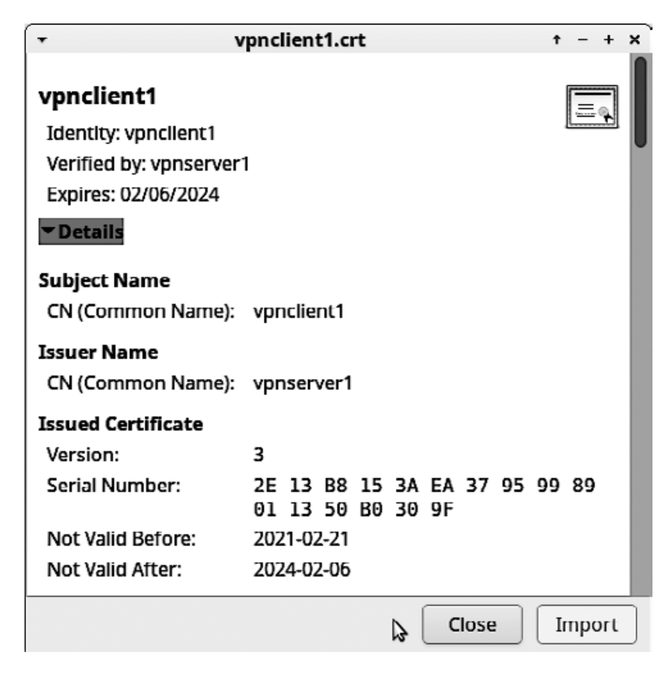


Рис. 13.1. Информация о подписанном сертификате

Ту же информацию можно получить с помощью команды `openssl`:

```
$ openssl x509 -noout -text -in vpnserver1.crt
```

Сертификат — это запрос, подписанный центром сертификации. Он содержит открытый ключ и цифровую подпись ЦС. Запрос содержит открытый ключ и цифровую подпись соответствующего закрытого ключа. Вы можете увидеть все это, сравнив их.

Инструментарий EasyRSA изначально был частью OpenVPN, а затем был выделен в отдельный проект. Если у вас есть опыт управления инфраструктурой PKI с помощью OpenSSL, то вы наверняка оцените, насколько EasyRSA упрощает этот процесс.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.6. Настройка параметров по умолчанию EasyRSA

Задача

Настройки по умолчанию EasyRSA вас не устраивают, и вам хотелось бы узнать, как их поменять.

Решение

Найдите у себя в системе файл `vars.example`, который является частью EasyRSA. Сохраните копию этого файла как `vars` в каталоге с инфраструктурой PKI, которым в примерах в данной главе является каталог `/home/duchess/mypki/pki/`. Файл `vars` определяет настройки по умолчанию, используемые при создании и подписании сертификатов.

Данный файл содержит подробные комментарии. Внесите свои изменения ниже строки `# DO YOUR EDITS BELOW THIS POINT` (вносите свои правки ниже этой строки). Все, что начинается с `set_var`, можно редактировать. Не забудьте раскомментировать свои изменения.

Например, в конфигурации по умолчанию используется только общее имя (`Common Name`), а не полная конфигурация организации `org`. В следующем примере создается традиционная конфигурация организации:

```
set_var EASYRSA_DN      "org"
set_var EASYRSA_REQ_COUNTRY    "US"
set_var EASYRSA_REQ_PROVINCE   "Oregon"
set_var EASYRSA_REQ_CITY       "Walla Walla"
set_var EASYRSA_REQ_ORG        "MyCo"
set_var EASYRSA_REQ_EMAIL      "me@example.com"
set_var EASYRSA_REQ_OU         "MyOU"
```

Используя конфигурацию организации, не забудьте ввести свое общее имя (`Common Name`) после запуска команды `easyrsa build-ca`, иначе будет использовано имя по умолчанию `Easy-RSA CA`:

```
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:myCN
```

Комментарий

Используйте `cn` или `org` в соответствии с вашими требованиями и предпочтениями; для работы сервера это не имеет значения.

См. рецепт 13.10, чтобы узнать, как упрочить защиту сервера.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.7. Создание и тестирование конфигураций сервера и клиента

Задача

После создания инфраструктуры открытых ключей (PKI) необходимо настроить сервер OpenVPN и клиентов.

Решение

В данном рецепте мы настроим простую тестовую конфигурацию с двумя хостами в одной подсети, `server1` и `client1`. Это хороший простой способ проверить настройки сервера, не беспокоясь о маршрутизации и пересечении вашего интернет-шлюза.

В следующем примере представлены настройки сервера OpenVPN. Обратите внимание, что серверные ключи можно хранить где угодно на сервере, важно только правильно указать их местоположение в файле с настройками:

```
# vpnserver1.conf
port 1194
proto udp
dev tun
user nobody
group nobody
ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/vpnserver1.crt
key /etc/openvpn/keys/vpnserver1.key
dh /etc/openvpn/keys/dh.pem
tls-auth /etc/openvpn/keys/ta.key 0

server 10.10.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
persist-key
persist-tun
```

```
tls-server
remote-cert-tls client

status openvpn-status.log
verb 4
mute 20
explicit-exit-notify 1
```

Пример настроек клиента:

```
# vpnclient1.conf
client
dev tun
proto udp
remote server1 1194

persist-key
persist-tun
resolv-retry infinite
nobind

user nobody
group nobody
tls-client
remote-cert-tls server
verb 4

ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/vpnclient1.crt
key /etc/openvpn/keys/vpnclient1.key
tls-auth /etc/openvpn/keys/ta.key 1
```

Остановите сервер OpenVPN, если он запущен:

```
$ sudo systemctl stop openvpn@.openvpn1.service
```

Запустите OpenVPN на обоих хостах с помощью команды `openvpn`:

```
$ sudo openvpn /etc/openvpn/vpnserver1.conf
Tue Feb 16 16:50:49 2021 us=265445 Current Parameter Settings:
Tue Feb 16 16:50:49 2021 us=265481 config = '/etc/openvpn/vpnserver1.conf'
[...]
Tue Feb 16 16:50:49 2021 us=270212 Initialization Sequence Completed

$ sudo openvpn /etc/openvpn/vpnclient1.conf
Tue Feb 16 16:56:22 2021 OpenVPN 2.4.3 x86_64-suse-linux-gnu [SSL (OpenSSL)]
[LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] built on Jun 20 2017
Tue Feb 16 16:56:22 2021 library versions: OpenSSL 1.1.1d 10 Sep 2019, LZO 2.10
Enter Private Key Password: *******
[...]
Tue Feb 16 16:56:26 2021 Initialization Sequence Completed
```

Вот и все! Вывод команд подтверждает, что настройки верны и между хостами успешно установлено соединение. Для остановки нажмите Ctrl+C на обоих хостах.

Комментарий

OpenVPN устанавливается с набором примеров конфигураций в `/usr/share/doc/openvpn/`. Они содержат множество комментариев и вполне могут служить справочными пособиями. Параметров десятки, но в реальной жизни вы будете использовать лишь часть из них. В примерах в этом рецепте есть несколько важных моментов, которые хотелось бы отметить особо.

`port 1194` — это порт по умолчанию, а `proto udp` предпочтительнее, чем `proto tcp`. Протокол UDP более безопасный, он обеспечивает некоторую защиту от сканирования портов и атак типа «отказ в обслуживании», а также имеет более высокую пропускную способность и меньшую задержку. TCP удобнее, когда удаленный пользователь заходит в общедоступные сети с ограничивающими брандмауэрами, такие как сети отелей или кафе.

Параметр `tls-auth /etc/openvpn/keys/ta.key` всегда должен иметь значение `0` на сервере и `1` на клиентах. Он обеспечивает подключение только по протоколу TLS.

`verb 4` — уровень подробности журналирования: 1 — самый низкий, 9 — самый подробный. Выбирайте уровни 4–6, пока не убедитесь, что все настроено правильно. С помощью этих уровней при запуске OpenVPN из командной строки вы увидите много сообщений.

Существует множество устаревших руководств, в которых рекомендуется использовать параметр `comp_lzo` для включения сжатия. Не обращайте внимания на эту рекомендацию, поскольку сжатие не принесет особой пользы. Большая часть трафика не сжимается, поскольку он либо уже сжат, либо зашифрован и не может быть сжат. Есть как минимум одна уязвимость, доступная при сжатии, VORACLE.

Дополнительная информация

- Примеры конфигурационных файлов в вашей системе.
- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.8. Управление OpenVPN с помощью команды systemctl

Задача

Вы хотите управлять демоном OpenVPN так же, как любым другим демоном, то есть с помощью команды `systemctl`, но вы не можете найти файл модуля OpenVPN. Или у вас есть файл модуля с каким-нибудь странным именем, например `openvpn-server@.service`, но, когда вы пытаетесь его запустить, он выводит сообщения об ошибках.

Решение

Символ `@` включается в имена файлов параметризованных модулей. Это означает, что вы можете создать несколько файлов модулей для одной и той же службы, использующей разные файлы конфигурации. Например, предположим, что настройки вашего сервера хранятся в файле `/etc/openvpn/austin.conf`. Тогда для их использования вы можете создать файл модуля `openvpn@austin.service` с помощью команды `systemctl`:

```
$ sudo systemctl enable openvpn@austin
Created symlink /etc/systemd/system/multi-user.target.wants/openvpn@austin.
service
→ /usr/lib/systemd/system/openvpn@.service.
Created symlink /etc/systemd/system/openvpn.target.wants/openvpn@austin.service
→ /usr/lib/systemd/system/openvpn@.service.
```

Обратите внимание, что в команде не нужно вводить расширение файла `.conf`. Теперь вы можете управлять своим демоном OpenVPN с помощью `systemctl`, как и любой другой службой.

Комментарий

Это просто гениальный метод, который позволяет создать нескольких конфигураций, обойдясь без необходимости писать несколько файлов модулей. Вы можете «параметризовать» любой файл модуля `systemd`.

Вы можете иметь несколько туннелей, работающих одновременно на одной машине. Каждая конфигурация требует отдельного устройства `tun`, например `tun0`, `tun1`, `tun2`, отдельной подсети для каждого туннеля и отдельного порта UDP. Управляйте всеми этими туннелями с помощью различных файлов конфигурации и соответствующих им файлов параметризованных модулей.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- systemd.unit (https://oreil.ly/2AAEe).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.9. Распространение конфигураций клиентов с помощью файлов .ovpn

Задача

Настройка клиентов — довольно сложная работа, и хотелось бы знать, есть ли более быстрый способ, который могли бы применять ваши пользователи без посторонней помощи.

Решение

Объедините конфигурации и ключи клиентов в отдельные файлы с расширением `.ovpn`. Эти файлы могут импортировать все клиенты — для Linux, Windows, macOS, iOS и Android.

Сначала создайте сертификаты пользователей, а затем следуйте описанному далее шаблону и создайте файлы `.ovpn`. Этот пример основан на рецепте 13.7. Чтобы не ссылаться на все сертификаты, скопируйте их в данный файл. Сертификаты представлены в обычном текстовом виде, поэтому вам нужно только скопировать блоки BEGIN/END в файл `.ovpn`:

```
# vpnclient1.ovpn
client
dev tun
proto udp
remote server2 1194

persist-key
persist-tun
resolv-retry infinite
nobind

user nobody
group nobody
```

```
tls-client
remote-cert-tls server
verb 4

# ca.crt
<ca>
-----BEGIN CERTIFICATE-----
MIIDSDCCAjCgAwIBAgIUD2UxdEwgvhhr0zq5fAxIDIueB2EwDQYJKoZIhvcNAQEL
BQAwFTETMBEGA1UEAwwKdnBuc2VydmVyMTAeFw0yMTAyMjExODU1MjNaFw0zMTAy
MTkxODU1MjNaMBUxEzARBgNVBAMMCnZwbnNlcnZlcjEwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQDpQJo+Izt8v0zriSWwrChc1tnVj3E3h3XuyEHub7hj
y4bMu2PqKByFNr+iikEF3u0d6HrCRSDKt1BcLzL3TsTJ/hJBHAlTyqEgVce1knjL
2g9NnDbekRtJSJCxS9j+RWtP43Xdg5edb5hTCZqdNFHD8oNuSMGFBbHN4oi9eDXl
rvyVHJe+UkI1Ow6mW0+ln/IoKNFPovz+l+ds3fJ5+UHe2TaQPQc7tGZ33j7wfJQd
es8baFdK+lnmGdUOrW9BQE6ReMSezkz6dKdIZdy7jEs6xoflOzyWlgydmnkAvLnx
MBQDgDUbc5MuooVMAWa4yhtz0B9ZmdJDb8jzHDpTPqdRAgMBAAGjgY8wgYwwHQYD
VR0OBBYEFF8KPhl1xxV0110JiBs5iUEPoJ1IMFAGA1UdIwRJMEeAFF8KPhl1xxV0
110JiBs5iUEPoJ1IoRmkFzAVMRMwEQYDVQQDDAp2cG5zZXJ2ZXIxghQPZTF0TCC+
GGvTOrl8DEgMi54HYTAMBgNVHRMEBTADAQH/MAsGA1UdDwQEAwIBBjANBgkqhkiG
9w0BAQsFAAOCAQEAMnRLz3CBApSrjfUKsWYioNGQGvh77Smh/1hPGIu4eEldQSmZ
Aj7qclEaORdBxmqrVtA3Z9cX1L0xFrg14nLyddmuWHG3ZChc5ZMpYtD2YpOH265B
FFjDp96vK13dpixWKrVpvakLCCA4EvnC8CEjbm0oNFiCgSwKAoJFCcUzwC33swsU
B2w5/iT6CZKuKhSmET1IDpG8krGC/Ib2GNAS0szMI94P0ajZgVznMcXOJ7gUg4rM
sEB8OzM6GBEZTqbAa9uVMZnOZvZA5jGIbBuelUo0bqGdAyx2B68zzuL//qvsHsvw
kZCyKIaXH0NBV7vexMKWcwFLLBzWizFQbbFpFA==
-----END CERTIFICATE-----
</ca>

# vpnclient1.cert
<cert>
-----BEGIN CERTIFICATE-----
MIIDVjCCAj6gAwIBAgIQLhO4FTrqN5WZiQETULAwnzANBgkqhkiG9w0BAQsFADAV
MRMwEQYDVQQDDAp2cG5zZXJ2ZXIxMB4XDTIxMDIyMTE4NTYzM1oXDTI0MDIwNjE4
NTYzM1owFTETMBEGA1UEAwwKdnBuY2xpZW50MTCCASIwDQYJKoZIhvcNAQEBBQAD
ggEPADCCAQoCggEBALUFYXwk6JW/hRtoMs0Ug5jMcWXsjMUsCz8L8CeXNOs3wQrf
YBWF1TYCLPd2/vwXsvbqCE85IZwjsJ5mEx9YgQ5M1teDkLZqBn8y7VIyDAAU8RsN
NcrnpeMDV0LgZIBeUrHi4ZTooaw4FdJ5BBYRHR1APVaaHDWx59ohJuBDpriWhvWk
lWX0rpSJltXriIOCzky/yEwfw6ah5jWaTgfe41fXq8j3lx2IbgIL7I4//jhC6JYz
N7huTdT2uB2MUbYX0XWBffMG8wcBZtMI2XryZmPvFYWP7N5nZZsBXkLz/UngAu3k
jkYJOnJy/hdOFLN/yXj7VFydmivUSeekdjjxyAECAwEAAaOBoTCBnjAJBgNVHRME
AjAAMB0GA1UdDgQWBBSnLIQoTPLyECbJHfgYBHvQpcmfgzBQBgNVHSMESTBHgBRf
Cj4ZdccVdNddCYgbOYlBD6CdSKEZpBcwFTETMBEGA1UEAwwKdnBuc2VydmVyMYIU
D2UxdEwgvhhr0zq5fAxIDIueB2EwEwYDVR0lBAwwCgYIKwYBBQUHAwIwCwYDVR0P
BAQDAgeAMA0GCSqGSIb3DQEBCwUAA4IBAQBaBpYZXVYUzOcXOVSaijmOZAIVBTeJ
meQz9xBQjqDXaRvypWlQ1gQtO8WnK9ruafc1g/h7LtvqtiALnGiJ0NbshkH8C1KE
yen46UCau5B/Xi0gA7FoPildvYdKSn/jI6KySCsplubjnJK9H/6DjAcEuqFLcsaY
5vpKQGP9Vl7H7hEVs4f1aory1T4Ma/bdXEOqgzHmIARLmxYeJm90sUT/n7e7VXfy
fILZ+8D1fMxCbeQRBkg1e8wJfgEbMRY9aGGt1qAs9gkm9RPelGB18v4iCbyebv3X
4hVHmfjcixdbWiABC7yq/gisooQ0robW/92dgemcwO0awHZX+opNBgwr
-----END CERTIFICATE-----
</cert>
```

```
# vpnclient1.key
<key>
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFHDBOBgkqhkiG9w0BBQ0wQTApBgkqhkiG9w0BBQwwHAQInjFvz5a4mY8CAggA
MAwGCCqGSIb3DQIJBQAwFAYIKoZIhvcNAwcECNsxQXxvMpN0BIIEyEZdgFwPnGup
vyhywXR6l6ihvHK2GRczIgH0mFIiwQDgDjZj2YsEnvSA/P3MHplkU/bgv9DJ5j2T
C5wPDmGN4yG1boHx9BQKbXqxGwdz/UcHwmNKur9qnSFrSVEvMDwvum+rmzWuKykf
gkKKBCT1JZ2DWKtjjDNYG9qhBn3S2zYVq311dDuLbBcruvo1UL031sDDYWTpVuuf
zZc0ozng0Nzb35bNkG6Ib+LYLzJi4stxzw0DTFl52lKv++R6xhmqb81IJE3vBs4H
DOutkYfifO1eGqEKksPQRl8n03UVkOtB5pH8VdQeLqEBBaq3qeIfU6FkH9XrPR/E
8VOg9BNpbyuUW7bQu0MzuJ8Ofkjy9K+HHdwFtGPyOatkeaXT/qcKVMvzWcbr8bPc
VncavzXdzo0Sb8FigsKYU1lNjgo00Phd3m0AOfptrweK6ucBds5SmqNrUFXiQ2JA
Ms3LUw4CXBBgvdu5TsA2xLGysip0RPKLyTnUPGnXxbBaaHMv8Jz3XRCrWgZbtAE3
XhE9fKw+ZMEP+2jpC/1mjN/N9VuJfYZEhgA84wzYMu6pt3zPkWZqR6yGTDFEDhvh
OAZYEpqrhe++nxDpuQlpCCl4IndSg9L9oX1ydrvPNHGbRVztd3+r9wr4Ub3fJ1g/
9ckCdanohEymKbjw34HEMmdx+fn5k2T9bLnl8fsYtcESkg04ChON3yOnZFKl6chT
BQ9X2Qmeg7FoawWiUY5o+7OHNKL7QpRt4jXPbXNuXFK9EYvuRzUqubLhL5DdmjuO
Se1vvZg7fT4C8qjYsoCa18idA00EN3ePFFf9AssHCoVW92GiUTTKG+qURCjtNtG6
dnPvxiSf98OBkkjeX3ni0cKdfMGoQTSdEy5GexvfRMF5HJrGO+CWXmqSBsuIlPUe
quqCsPmpaT2Ws/0UU9cKe4qaKjTL7CghtFmUEhH7t6Cd41Ki9gKi33j354l9w7l
J1bgca4rRUCecp2BPF3IjJc/RnTvHkbUK4mDX9s8xJhYf9WE6JYsk3NBSNNIj/9G
FMJlo71x8H3OAdFzRN5bjV797HByZ+YidZIgGAx2dSko3PQPy7RSxdmzFbxfUvzj
9jcYEu+V9unbtDK2qZ9I+LqXGE+EXjPBui40IWp8XIYNlSLn2qgroH079lXhXKBY
+DzcBzyT7GTX2QeYE+yqqPRIFWHnbnsnD6dMnAa46h+Si+f5sq33rfRsF7UpK4gV
IhzFkncCM47/Taqi0OY04Q40LuSCDjmjFL+VzZOsAtWGRNYNzIgniThEehElJwfI
ErzClcVptjhtCer8BPuO7YaMIHk1hKecHFqw3RrimWzroL1iu9Q29m2oM+bVc6mD
we6r+t8JbaAFxoHBK4i6M0rcdJPICxDTIOjPC3Fg/MeqiCi7F0DFZvXwPGRD+0Of
MBnsDplEUjK06jbE5BjGQ7n7P+dwDxyp/aVO4CfX7ZOco6h9r3b6nqlzPVNE9erw
kS7WwT/TWraw/sfIO9sNSgle7PoRh2s/w/oGVhC6ymlMdXe+mhMzHFnGEbBRh2Rd
kd/EdYNubHg0k9+RLTwbgwZ+176cIJyOpqaoJGv0bsKM8X26Pk/fkyF6xgdQYQOx
8i9Whea8OjUOQAcgc7gUyA==
-----END ENCRYPTED PRIVATE KEY-----
</key>

# ta.key
<tls-auth>
-----BEGIN OpenVPN Static key V1-----
4eb35b44d1d8a82cfa51af394d4f58f3
69bf8fe8c0a0a032f38b0ee104889628
8a5dc89486736b39d64ad3c6831bf9ba
9f3f96c3307d322a5bf055b9bc3bfa74
929faf361c14de97445f5927794264bb
e3f71c925f2236cfb0109ecfd6406cef
857dfb39783a09ecd56d3cf09ebbc853
0f43b1c787f0db99dbecabcd2090cfbb
54c86d8102a5430fd6a7f37ab5ce8ed9
f6bec8984bde4267f78913ff702dd396
a205b6be9e7ab41cf1ebad3953c27c7c
f3b435345e02aede049ef7c9f1c2704f
2ed91110ccb19d0d3bd46a00f54c73e2
07b31160cdc54c3f5a7989bb999ac5f3
89c6de7e79fc93399924a8d298eab462
```

```
231234e690c319d5cbd832788f0dbcfb
-----END OpenVPN Static key V1-----
</tls-auth>
```

Теперь вам нужно распространить среди клиентов только один файл. (См. рецепт 13.1, чтобы узнать, какое программное обеспечение необходимо устанавливать для клиентов Linux, macOS, Windows, iOS и Android.)

Самый простой способ импортировать новый файл `.ovpn` в Linux — использовать NetworkManager. Откройте настройки NetworkManager, нажмите кнопку Add new connection (Добавить новое соединение). В открывшемся окне Choose a VPN Connection Type (Выбор типа подключения) выберите пункт Import a Saved VPN Connection (Импортировать VPN-соединение), нажмите кнопку Create (Создать) и найдите файл `.ovpn` в диалоге выбора файлов. Проверьте настройки на вкладках General (Основные параметры) и VPN.

Убедитесь, что на вкладке General (Основные параметры) снят флажок All users may connect to this network (Все пользователи могут подключаться к этой сети). Это простая, но важная мера предосторожности, которая требует, чтобы у каждого пользователя была своя отдельная конфигурация OpenVPN.

На вкладке VPN обратите внимание, что NetworkManager преобразует встроенные сертификаты в файлы `.pem` (рис. 13.2). Так и должно быть. При желании можете сравнить их с оригиналами — щелкните на маленьком значке с изображением папки справа от поля, чтобы увидеть, где хранятся преобразованные файлы.

Для всех других клиентов процедура импортирования выглядит аналогично. Следуйте их инструкциям, и если не произойдет ничего экстраординарного, то ваши клиенты будут готовы к работе через пару минут.

Функция импорта в NetworkManager работает также с конфигурационными файлами клиента, не включающими ключи и сертификаты, как было показано в рецепте 13.7. В этом случае сертификаты не преобразуются и сохраняются со своими оригинальными именами файлов.

Объединенный файл может иметь расширение `.conf`, если у вас только клиенты Linux.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).



Рис. 13.2. Импортирование файла .ovpn с настройками клиента в NetworkManager

13.10. Повышение безопасности сервера OpenVPN

Задача

Хотелось бы узнать, как еще больше обезопасить сервер OpenVPN.

Решение

Настройки OpenVPN по умолчанию достаточно хороши, но они ориентированы на максимальную совместимость. Вы можете внести в них ряд корректив, которые еще больше обезопасят ваш сервер.

Следующие примеры демонстрируют настройки, которые можно добавлять в конфигурационные файлы и серверов, и клиентов. Эти параметры максимизируют эффективность TLS. Все протоколы SSL и TLS старше TLS 1.2 объявлены устаревшими и должны быть отключены. Принимайте только TLS 1.2 или выше:

```
tls-version-min 1.2
tls-version-max 1.3 or-highest
```

Задействуйте более надежный алгоритм шифрования канала данных и принудительно используйте его, отключив согласование шифрования:

```
AES-128-GCM
ncp-disable
```

В TLS 1.3 появилось много изменений, поэтому вам понадобятся две разные конфигурации для TLS 1.2 и 1.3. Все алгоритмы шифрования, перечисленные ниже, более эффективны:

```
# TLS 1.3
tls-ciphersuites TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
# TLS 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-CHACHA20-POLY1305-SHA256:TLS-ECDHE-RSAWITH-
CHACHA20-POLY1305-SHA256:TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256:
TLS-ECDHE-RSA-WITH-AES-128-GCM-SHA256
```

Используйте протокол Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) вместо статических ключей Диффи — Хеллмана (Diffie — Hellman). Это избавит от необходимости создавать `ta.key`, как в рецепте 13.5:

```
dh none
ecdh-curve secp384r1
# используйте tls-server на сервере и tls-client на клиенте
tls-server
```

Добавьте параметр `float` (только в конфигурацию сервера), чтобы позволить клиентам перемещаться по разным сетям без потери соединения, если они успешно проходят все остальные этапы аутентификации.

Параметр `opt-verify` (предназначен только для конфигурации сервера) проверяет совместимость настроек сервера и клиента и отключает клиентов, которые несовместимы. Он проверяет настройки `dev-type`, `link-mtu`, `tun-mtu`, `proto`, `ifconfig`, `comp-lzo`, `fragment`, `keydir`, `cipher`, `auth`, `keysize`, `secret`, `no-replay`, `no-iv`, `tls-auth`, `key-method`, `tls-server` и `tls-client`.

Полный пример конфигурации вы найдете ниже в подразделе «Комментарий».

Комментарий

Включите все эти параметры в конфигурацию сервера:

```
# vpnserver1.conf
port 1194
proto udp
dev tun
user nobody
group nobody

ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/vpnserver1.crt
key /etc/openvpn/keys/vpnserver1.key

server 10.10.0.0 255.255.255.0
ifconfig-pool-persist ipp.txt
keepalive 10 120
persist-key
persist-tun
tls-server

remote-cert-tls client
verify-client-cert require
tls-cert-profile preferred
tls-version-min 1.2
tls-version-max 1.3 or-highest

float
opt-verify
AES-128-GCM
ncp-disable
dh none
ecdh-curve secp384r1

# TLS 1.3
tls-ciphersuites TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
# TLS 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-CHACHA20-POLY1305-SHA256:TLS-ECDHE-RSA-
WITH-CHACHA20-POLY1305-SHA256:TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256:
TLS-ECDHE-RSA-WITH-AES-128-GCM-SHA256

status openvpn-status.log
verb 4
mute 20
explicit-exit-notify 1
```

Пример конфигурации клиента с использованием формата объединенного файла (см. рецепт 13.9):

```
# vpnclient1.conf
client
dev tun
proto udp
remote server1 1194

persist-key
persist-tun
resolv-retry infinite
nobind

user nobody
group nobody
tls-client
remote-cert-tls server
verb 4

# Встроенные ключи
# ca.crt
<ca>
[...]
</ca>

# client.crt
<cert>
[...]
</cert>

# client.key
<key>
[...]
</key>

tls-version-min 1.2
tls-version-max 1.3 or-highest
AES-128-GCM
ncp-disable
dh none
ecdh-curve secp384r1

# Настройки шифрования для TLS 1.3
tls-ciphersuites TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
# Настройки шифрования для TLS 1.2
tls-cipher TLS-ECDHE-ECDSA-WITH-CHACHA20-POLY1305-SHA256:TLS-ECDHE-RSA-
WITHCHACHA20-
POLY1305-SHA256:TLS-ECDHE-ECDSA-WITH-AES-128-GCM-SHA256:TLS-ECDHE-RSAWITH-
AES-128-GCM-SHA256
```

```
status openvpn-status.log
verb 4
mute 20
explicit-exit-notify 1
```

Эти параметры усиливают аутентификацию между клиентом и сервером и гарантируют использование протокола TLS версии 1.2 и выше. Если вам это интересно, то, чтобы узнать, какие шифры и алгоритмы шифрования более предпочтительны, я опросила несколько экспертов. Вы можете обезопасить себя еще больше, например запретив пользователям сохранять пароли, добавив более строгую аутентификацию «клиент — сервер», применив SELinux или chroot. Однако все это довольно сложные темы, и здесь они рассматриваться не будут.

Дополнительная информация

- EasyRSA (https://oreil.ly/eKbsg).
- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`
- Сборник рецептов OpenSSL Cookbook (https://oreil.ly/Ctm0X).

13.11. Настройка сети

Задача

Сервер OpenVPN запущен, все проверки подключения выполнены успешно, и теперь хотелось узнать, как настроить сеть, чтобы удаленные клиенты могли найти ваш сервер и трафик маршрутизировался соответствующим образом.

Решение

Не существует универсального решения. При настройке виртуальной сети необходимо учитывать настройки локальной сети, предполагается ли подключать отдельных клиентов или целые сети, какой протокол используется: IPv4 или IPv6, как настроен интернет-шлюз и многое другое. Обратитесь к замечательной книге *OpenVPN Cookbook* (O'Reilly) Яна Джаста Кейсера (Jan Just Keijser), чтобы найти ответы на свои вопросы. В этой книге рассказывается об устройствах

TUN и TAP, клиентах Windows, PAM и LDAP, IPv6, маршрутизации и конфигураций типа «сеть — сеть».

Комментарий

Сеть — самая сложная часть в работе любого сервера, особенно для его безопасности. Поэтому стоит изучить данный вопрос и постараться сделать все правильно. В документации для OpenVPN также есть много полезной информации о сетях.

Дополнительная информация

- Документация для OpenVPN (https://oreil.ly/Ah124).
- `man 8 openvpn`

ГЛАВА 14

Создание брандмауэра на основе firewalld

В этой главе охватываются основы использования firewalld для создания брандмауэра хоста. У разных хостов разные требования. Например, сервер должен разрешать различные типы входящих запросов на подключение, а персональный компьютер, на котором не работают службы, обслуживающие клиентов, не должен принимать запросы на подключение. Переносной компьютер, используемый для доступа к нескольким сетям, должен давать возможность динамически управлять брандмауэром.

Обзор firewalld

firewalld, как и все брандмауэры (межсетевые экраны, файерволы), имеет очень длинный список возможностей. В основном мы будем учиться использовать *зоны* для управления входящим трафиком. Зона — это контейнер для уровня доверия; например, одни зоны разрешают всевозможные входящие запросы на соединение, а другие, напротив, сильно ограничивают перечень таких запросов. Каждому сетевому интерфейсу в системе может быть назначена только одна зона, и одна зона может быть назначена нескольким интерфейсам.



Требуются знания устройства сети

Наиболее важные сетевые понятия, которые необходимо знать, — это порты, службы, TCP, UDP, переадресация портов (port forwarding), маскарадинг (masquerade), маршрутизация и IP-адресация. Когда вы освоите эти понятия, вы поймете, как настроить брандмауэр. Если вам нужно руководство по компьютерным сетям, то прочтите книгу Networking Fundamentals Гордона Дэвиса (Gordon Davies; Packt Publishing) или Networking All-in-One For Dummies Дуга Лоу (Doug Lowe, For Dummies). Если у вас есть подписка на O'Reilly Learning Platform (https://oreil.ly/mEsNB), то там вы тоже сможете найти массу полезной информации.

Традиционный брандмауэр в Linux основан на инфраструктуре фильтрации пакетов *netfilter* в ядре Linux, которая фильтрует входящий и исходящий сетевой трафик, и *iptables* — программном обеспечении, используемом для создания таблиц с правилами фильтрации трафика и управления ими.

Времена меняются, и на смену iptables пришли новые диспетчеры правил, такие как *ufw* (uncomplicated firewall — «несложный брандмауэр»), *nftables* (Netfilter tables — «таблицы Netfilter») и *firewalld* (firewall daemon — «демон брандмауэра»). Последний, так же как iptables и nftables, использует таблицы с правилами для управления фильтрацией трафика. Он имеет интерфейс командной строки и красивый графический интерфейс *firewallconfig*. firewalld — это интерфейс как для iptables, так и для nftables. nftables имеет значительные усовершенствования, по сравнению с iptables, и предназначен для применения в качестве базовой поддержки firewalld по умолчанию, но в некоторых дистрибутивах Linux по-прежнему по умолчанию используется iptables. Выбор между iptables и nftables производится с помощью параметра `FirewallBackend` в `/etc/firewalld/firewalld.conf` (см. рецепт 14.4).

firewalld распространяется с предопределенными наборами правил, которые называются *зонами* и предназначены для разных случаев использования, таких как компьютер, на котором нет сетевых служб, компьютер, на котором запущены службы, и разные зоны для разных сетевых интерфейсов на одном компьютере. Эти зоны можно редактировать, настраивая их в соответствии со своими требованиями.

Зоны в firewalld управляют *службами* — конфигурациями для распространенных служб, таких как ssh, imaps и rsync. Большинство предопределенных служб поддерживают только стандартные назначения портов. Но вы можете редактировать их по своему усмотрению и создавать собственные настраиваемые зоны.

firewalld интегрирован с NetworkManager, поэтому вам не придется беспокоиться об управлении динамическими подключениями, например, когда вы носите свой ноутбук с собой и подключаетесь к разным сетям.



Служба NetworkManager

NetworkManager — важная часть Linux с 2004 года. Она заменила мешанину громоздких сетевых клиентских инструментов и управляет всеми сетевыми интерфейсами и сетевыми соединениями. Если вы не знакомы с NetworkManager, то обращайтесь на сайт проекта GNOME NetworkManager (https://oreil.ly/hkqaq).

Если вы эксплуатируете общедоступные серверы, используя коммерческую услугу хостинга, то настройка вашего брандмауэра будет зависеть от того, какие возможности поддерживает ваш поставщик услуг. Защита общедоступных серверов, таких как веб-серверы и интернет-магазины, независимо от того,

размещены они удаленно или в вашем собственном центре обработки данных, требует навыков и умений, обсуждение которых выходит за рамки данной книги. Поэтому вам лучше пройти углубленное обучение или нанять экспертов.

Как работают брандмауэры

Когда-то давно Ubuntu Linux поставлялся без брандмауэра, поскольку при установке с настройками по умолчанию не запускались никакие сетевые службы и, следовательно, сетевые порты не прослушивались. А когда система не прослушивает порты, то атаковать ее бессмысленно. К счастью, в более поздних выпусках это решение было изменено, поскольку пользователи часто что-то меняют и даже самые опытные из них допускают ошибки, а злоумышленники всегда обнаруживают новые уязвимости. Безопасность — многоуровневый процесс.

Посмотрим, как работают брандмауэры. Основной принцип: запретить все, разрешить только необходимое.

Сетевая служба, например SSH-сервер, должна открыть сетевой порт, чтобы позволить удаленным пользователям входить в систему. Запуская такую службу, вы позволяете другим заходить в вашу систему. По умолчанию `sshd` прослушивает TCP-порт 22. Увидеть все прослушиваемые порты в системе можно с помощью команды `netstat`. Следующий фрагмент вывода этой команды показывает, как выглядит порт SSH:

```
$ sudo netstat -untap | sed '2p;/ssh/!d'
Proto Recv-Q Send-Q Local Address   Foreign Address  State   PID/Program name
tcp        0      0 0.0.0.0:22      0.0.0.0:*        LISTEN  1296/sshd: /usr/sbi
tcp6       0      0 :::22           :::*             LISTEN  1296/sshd: /usr/sbi
```

Этот пример показывает, что на текущий момент нет активных соединений, поскольку все поля Foreign Address (Внешний адрес) равны нулю, а поле State (Состояние) содержит слово LISTEN (Прослушивается). `sshd` прослушивает входящие соединения IPv4 и IPv6 на всех сетевых интерфейсах и всех IP-адресах на TCP-порту 22. Комбинация IP-адреса и номера порта — это адрес, сообщающий ядру Linux, куда отправлять SSH-пакеты.

Далее показан пример активного соединения SSH с состоянием ESTABLISHED (Установлено). В нем указаны локальный адрес и порт, к которым подключена удаленная машина, а также внешний адрес и порт удаленной машины (столбцы `Recv+Q` и `Send+Q` были опущены для ясности):

```
$ sudo netstat -untap | sed '2p;/ssh/!d'
Proto  Local Address     Foreign Address     State       PID/Program name
tcp    0.0.0.0:22        0.0.0.0:*           LISTEN      1296/sshd: /usr/sbi
tcp    192.168.1.97:22   192.168.1.91:56142  ESTABLISHED 13784/sshd: duchess
tcp6   :::22             :::*                LISTEN      1296/sshd: /usr/sbi
```

Есть несколько способов указать, какие пакеты TCP/IP можно передать по определенному IP-адресу и в определенный порт. Большинство серверов настроены для прослушивания только определенных сетевых интерфейсов или IP-адресов и для приема запросов с определенных адресов и диапазонов адресов. Брандмауэр добавляет дополнительные элементы управления, и желательно использовать оба средства управления.

Сетевые порты и их нумерация

В системе Linux имеется 65 536 возможных сетевых портов, пронумерованных от 0 до 65 535, и многие из них зарезервированы для определенных служб. Порт с номером 0 вообще не используется. Все зарезервированные номера портов можно увидеть в файле `/etc/services`, который имеется в каждом дистрибутиве Linux. Полный официальный список ищите в реестре номеров портов, имен служб и транспортных протоколов IANA (https://oreil.ly/CF0bF).

Диапазоны номеров портов организованы следующим образом:

- 0–1023 — *общеизвестные порты*. Это системные порты для распространенных служб, таких как FTPS (безопасный обмен файлами), SSH (безопасный удаленный вход в систему), NTP (протокол сетевого времени), POP3 (электронная почта), HTTPS (порт безопасного веб-сервера) и т. д.;
- 1024–49 151 — *зарегистрированные порты*, предназначенные для дополнительных служб;
- 49 152–65 535 — *временные порты*, которые также называют частными и динамическими. Они используются системой для завершения установки соединений с удаленными службами. Например, когда вы занимаетесь веб-серфингом, в выводе команды `netstat` это выглядит так (столбцы `Recv+Q` и `Send+Q` опущены для ясности):

```
$ sudo netstat -untap
Proto  Local Address         Foreign Address    State        PID/Program name
[...]
tcp    192.168.43.234:50586  72.21.91.66:443    ESTABLISHED  2798/firefox
tcp    192.168.43.234:38262  52.36.174.147:443  ESTABLISHED  6481/chrome
tcp    192.168.43.234:53232  99.86.33.45:443    ESTABLISHED  2798/firefox
[...]
```

Этот пример иллюстрирует ответ на исходящий запрос с вашего компьютера. Посещая сайт, вы инициируете запрос на подключение, и удаленный веб-сервер отправляет ответы на временные сетевые порты вашей системы. Первое соединение в списке установлено с локальным компьютером с IP-адресом 192.168.43.234 и портом 50 586. Внешний адрес (колонка Foreign Address) — это IP-адрес и порт удаленного сервера. Состояние `ESTABLISHED` означает, что соединение с удален-

ной машиной установлено. По завершении сеанса после закрытия браузера порт 50 586 освободится и будет готов к повторному использованию.

Временные порты не прослушиваются службами. Соединения с временными портами являются временными и создаются только в ответ на исходящий запрос на подключение с вашего компьютера, например, при посещении сайта. Брандмауэр может блокировать временные порты, но тогда у вас не будет доступа к хостам или сайтам за пределами вашего компьютера.

14.1. Определение того, какой брандмауэр запущен

Задача

Узнать, какой брандмауэр используется в вашей системе Linux.

Решение

Начните с документации для вашего дистрибутива Linux, так как большинство дистрибутивов включают брандмауэр по умолчанию. Наиболее распространены: *iptables* (Internet Protocol tables), *ufw* (Uncomplicated Firewall) и *nftables* (Netfilter tables). Все три брандмауэра основаны на использовании правил фильтрации в инфраструктуре netfilter, которая является частью ядра Linux.

Затем посмотрите, что говорит systemd. Следующий пример показывает, что в данной системе используется nftables:

```
$ systemctl status nftables.service
    ● nftables.service - Netfilter Tables
     Loaded: loaded (/usr/lib/systemd/system/nftables.service; disabled; vendor>
     Active: active (exited) since Sat 2020-10-17 13:15:05 PDT; 4s ago
       Docs: man:nft(8)
    Process: 3276 ExecStart=/sbin/nft -f /etc/sysconfig/nftables.conf (code=exi>
   Main PID: 3276 (code=exited, status=0/SUCCESS)
   [...]
```

Следующий пример показывает, что запущен firewalld:

```
$ systemctl status firewalld.service
● firewalld.service - firewalld - dynamic firewall daemon
Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor>
     Active: active (running) since Sat 2020-10-17 12:36:20 PDT; 37min ago
       Docs: man:firewalld(1)
   Main PID: 775 (firewalld)
      Tasks: 2 (limit: 4665)
     Memory: 40.9M
     [...]
```

Следующий пример проверяет ufw и показывает, что этот брандмауэр установлен, но неактивен:

```
$ systemctl status ufw.service
● ufw.service - Uncomplicated firewall
     Loaded: loaded (/lib/systemd/system/ufw.service; disabled; vendor preset:
enabled)
     Active: inactive (dead)
       Docs: man:ufw(8)
```

Если какой-то из них не установлен, то вы увидите соответствующее сообщение.

Вы можете удалить ufw и nftables или замаскировать их, чтобы они не запускались:

```
$ sudo systemctl stop ufw.service
$ sudo systemctl mask ufw.service

$ sudo systemctl stop nftables.service
$ sudo systemctl mask nftables.service
```

Комментарий

Запускайте только один брандмауэр, если только вы не любитель распутывать конфликты между правилами брандмауэров.

Дополнительная информация

- Глава 4.
- https://firewalld.org.

14.2. Установка firewalld

Задача

Установить брандмауэр firewalld в систему Linux.

Решение

Если firewalld отсутствует в вашей системе, то установите пакет *firewalld*, а также пакет *firewall-config*, чтобы получить графический интерфейс для настройки.

Комментарий

Пока, к счастью, все основные дистрибутивы Linux используют одни и те же имена пакетов, firewalld и firewall-config.

После установки firewalld может запуститься или не запуститься автоматически, в зависимости от вашего дистрибутива Linux. Но создание и тестирование правил требует, чтобы он был запущен.

Если возможно, то отключите сетевое соединение своего компьютера до завершения первоначальной настройки firewalld. Например, щелкните на значке апплета NetworkManager, который по умолчанию устанавливается и используется в большинстве дистрибутивов Linux (рис. 14.1).



Рис. 14.1. Отключение от сети с помощью NetworkManager

Или воспользуйтесь утилитой *nmcli*. Ниже показано, как найти и отключить Wi-Fi-соединение. Используйте в команде отключения имя соединения из столбца `CONNECTION` (Соединение):

```
$ nmcli device status
DEVICE  TYPE  STATE        CONNECTION
wlan0   wifi  connected    ACCESS_POINTE

$ nmcli connection down ACCESS_POINTE
Connection 'ACCESS_POINTE' successfully deactivated
(D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/4)
```

Чтобы восстановить соединение, выполните команду:

```
$ nmcli connection up ACCESS_POINTE
Connection successfully activated
(D-Bus active path: /org/freedesktop/NetworkManager/ActiveConnection/7)
```

Управление брандмауэром firewalld осуществляется как обычно — с помощью systemd. Ниже представлены основные команды:

- `systemctl status firewalld.service`;
- `sudo systemctl enable firewalld.service`;
- `sudo systemctl start firewalld.service`;
- `sudo systemctl stop firewalld.service`;
- `sudo systemctl restart firewalld.service`.

Дополнительная информация

- Глава 4.
- https://firewalld.org.
- Приложение (в конце книги).

14.3. Определение номера установленной версии firewalld

Задача

Определить номер установленной версии firewalld.

Решение

Узнать установленную версию можно с помощью диспетчера пакетов или команды `firewall-cmd`:

```
$ sudo firewall-cmd --version
0.9.3
```

Комментарий

Чтобы узнать версию с помощью команды `firewall-cmd`, брандмауэр firewalld должен быть запущен. Иначе вы просто увидите сообщение FirewallD is not running (FirewallD не запущен).

Дополнительная информация

- https://firewalld.org.

14.4. Настройка iptables или nftables в роли базовой поддержки firewalld

Задача

Выбрать iptables или nftables в качестве базовой поддержки firewalld.

Решение

Отредактируйте файл `/etc/firewalld/firewalld.conf` в соответствии со своими предпочтениями:

```
FirewallBackend=nftables
```

или:

```
FirewallBackend=iptables
```

Затем перезапустите firewalld.

Комментарий

Вам может потребоваться установить пакет с iptables или nftables.

Если вы не определились в своем выборе, то выбирайте nftables, поскольку эта базовая поддержка активно развивается разработчиками firewalld.

Дополнительная информация

- Раздел «Обзор firewalld» в начале этой главы, на с. 389.
- Статья разработчиков firewalld в блоге nftables (https://oreil.ly/xO5eS) содержит подробную информацию об iptables и nftables и будущем развитии.

14.5. Вывод списка всех зон и всех служб, управляемых каждой зоной

Задача

Увидеть все зоны, определяемые конфигурацией firewalld, и службы, которыми управляют эти зоны.

Решение

Зона по умолчанию:

```
$ firewall-cmd --get-default-zone
public
```

Список всех зон:

```
$ firewall-cmd --get-zones
block dmz drop external home internal public trusted work
```

Список всех активных зон — зон, используемых в текущий момент:

```
$ firewall-cmd --get-active-zones
internal
  interfaces: eth1
work
  interfaces: wlan0
```

Настройки одной зоны:

```
$ sudo firewall-cmd --zone=public --list-all
public
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: dhcpv6-client ipp ipp-client mdns ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Настройки всех зон:

```
$ sudo firewall-cmd --list-all-zones
[...]
```

Комментарий

Зоны firewalld определяют уровни доверия для сетевых подключений. Каждая зона содержит описание и другие параметры, как показано выше на примере зоны *public*. Настройки зон определяются в файлах в формате XML, имена которых должны иметь расширение `.xml`. Загляните в `/usr/lib/firewalld/zone`, чтобы увидеть файлы с примерами.

Ниже представлены параметры зон:

- `target` — определяет действие по умолчанию для пакетов, не соответствующих ни одному правилу. Принимает одно из четырех значений: `default`, `ACCEPT`, `DROP` и `REJECT`. Например, когда в зону public с настройками, показанными в файле примера, поступают запросы на соединение для `dhcpv6-client`, `ipp`, `ipp-client`, `mdns` или `ssh`, то они принимаются. Любые пакеты, адресованные неразрешенным службам, отклоняются действием `default`, и отправителю посылается сообщение об отклонении;
- `ACCEPT` — принимает все пакеты, которые явно не заблокированы правилами;
- `DROP` — просто отбрасывает все пакеты, не разрешенные явно;
- `REJECT` — действует подобно `DROP`, но дополнительно посылает отправителю запроса сообщение об отклонении;
- `icmp-block-inversion` — инвертирует настройки для ICMP-запросов. Любые заблокированные запросы разблокируются, а разблокированные — блокируются. Обычно этому параметру присваивается значение `no` (нет);
- `interfaces` — определяет сетевой интерфейс или интерфейсы, к которым применяется эта зона. Каждый интерфейс можно привязать только к одной зоне, но одну и ту же зону можно применить к нескольким интерфейсам;
- `source` — задает IP-адреса, MAC-адреса и диапазоны IP-адресов. Например, с помощью этого параметра можно разрешить прием пакетов только из локальной сети, от определенных хостов, а также блокировать хосты или сети;
- `services` — список служб, управляемых данной зоной;
- `ports` — список номеров портов, управляемых этой зоной;

- `protocols` — список дополнительных протоколов TCP, управляемых этой зоной; протоколы задаются в том же виде, в каком они указаны в `/etc/protocol`;
- `masquerade` — либо `yes` (включить маскарадинг), либо `no` (выключить маскарадинг). Маскарадинг предназначен для совместного использования общего подключения к Интернету IPv4 хостами в локальной сети. Установите значение `no` на всех хостах, кроме маршрутизаторов;
- `forward-ports` — настраивает переадресацию пакетов с одного порта на другой;
- `source-ports` — перечисляет порты отправителя;
- `icmp-blocks` — перечисляет типы ICMP-запросов, которые нужно блокировать;
- `rich rules` — ваши собственные правила.

Дополнительная информация

- https://firewalld.org.
- `man 5 firewalld.zone`
- `man 1 firewall-cmd`

14.6. Ввод списка поддерживаемых служб

Задача

Получить список служб, поддерживаемых брандмауэром firewalld.

Решение

Используйте команду `firewall-cmd`:

```
$ sudo firewall-cmd --get-services
RH-Satellite-6 amanda-client amanda-k5-client amqp amqps apcupsd audit bacula
bacula-client bb bgp bitcoin bitcoin-rpc bitcoin-testnet bitcoin-testnet-rpc
bittorrent-lsd ceph ceph-mon cfengine cockpit condor-collector ctdb dhcp dhcpv6
[...]
```

Эта команда выводит довольно большой «кирпич» текста. Преобразовать его в более удобочитаемый список с одной колонкой можно следующим образом:

```
$ sudo firewall-cmd --get-services| xargs -n1
RH-Satellite-6
amanda-client
amanda-k5-client
amqp
amqps
apcupsd
[...]
```

Чтобы создать больше колонок, передайте параметр `xargs -n2`, `xargs -n3` и т. д.

Службы в firewalld могут иметь более сложные определения, чем простые адреса портов. Например, служба *bittorrent-lsd* включает два IP-адреса назначения:

```
$ sudo firewall-cmd --info-service bittorrent-lsd
bittorrent-lsd
  ports: 6771/udp
  protocols:
  source-ports:
  modules:
  destination: ipv4:239.192.152.143 ipv6:ff15::efc0:988f
  includes:
  helpers:
```

Служба *ceph-mon* открывает два порта для приема запросов:

```
$ sudo firewall-cmd --info-service ceph-mon
ceph-mon
  ports: 3300/tcp 6789/tcp
[...]
```

Вы можете редактировать настройки любых предопределенных служб в соответствии со своими потребностями.

Комментарий

Добавляя службы в зону, используйте их имена в точности так, как они указаны в списке. Вы можете создать свою службу, как описывается в разделе Add a Service в документации firewalld (https://oreil.ly/kvMYY).

Дополнительная информация

- https://firewalld.org.
- Раздел Add a Service в документации firewalld (https://oreil.ly/kvMYY).

14.7. Выбор и настройка зоны

Задача

Узнать, как правильно выбрать и настроить зону.

Решение

Выбор зоны firewalld зависит от того, какие службы работают на вашем компьютере. Если у вас не запущены никакие сетевые службы и требуется только обеспечить возможность подключения по сети, то используйте зону *drop* или *block*. Зона drop — наиболее строгая, она сбрасывает все входящие запросы на подключение и пропускает только ответы на запросы, отправленные с этого компьютера. Зона block похожа на drop, но дополнительно посылает отправителям сообщения об отказе.

Другие зоны в разных дистрибутивах Linux настраиваются по-разному, поэтому вам следует заглянуть в их настройки. Ниже представлен пример настроек зоны *work*:

```
$ sudo firewall-cmd --zone=work --list-all
work
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: dhcpv6-client ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Зону нужно привязать к сетевому интерфейсу. Например, следующие команды привяжут зону *work* к интерфейсу eth0 и затем проверят, как прошла привязка:

```
$ sudo firewall-cmd --zone=work --permanent --change-interface=eth0
success

$ sudo firewall-cmd --zone=work --list-interfaces
eth0
```

Если вы предпочитаете протестировать изменения, прежде чем сделать их постоянными, то опустите параметр `--permanent`. В результате будет создана

временная действующая конфигурация, которая немедленно вступит в силу. Временные изменения будут потеряны после перезапуска firewalld или выполнения команды `firewall-cmd --reload`. Чтобы этого не произошло, преобразуйте временные изменения в постоянные:

```
$ sudo firewall-cmd --runtime-to-permanent
```

После привязки зоны к сетевому интерфейсу нет необходимости перезагружать конфигурацию или перезапускать firewalld.

Комментарий

Как узнать, какую зону выбрать? Ниже перечислены зоны, предопределенные в Ubuntu 20.04, в порядке от наиболее строгих к наименее строгим. В вашем дистрибутиве зоны могут быть настроены немного иначе; см. рецепт 14.5, чтобы узнать, как просмотреть настройки вашей зоны.

- Зона *drop* — все входящие сетевые пакеты, которые не были запрошены вами, просто отбрасываются и отправителю не посылается никаких уведомлений. Пропускаются только входящие пакеты, которые передаются через соединения, инициированные с вашего компьютера. Это самая надежная защита при подключении к ненадежной сети, когда не требуется разрешать входящие SSH-подключения, обращения к общим файлам или любые другие запросы на подключение извне.
- Зона *block* — любые входящие запросы на подключение отклоняются сообщением `icmp-host-hibited` в сетях IPv4 и `icmp6-adm-prohibited` в сетях IPv6. Пропускаются только входящие пакеты, которые передаются через соединения, инициированные с вашего компьютера.
- Зона *public* — входящие соединения dhcpv6-client, ipp, ipp-client, mdns и ssh принимаются, а все остальные блокируются.
- Зона *external* — простой интернет-шлюз, сочетающий брандмауэр и простую маршрутизацию. Принимаются только входящие SSH-соединения, а осуществляется маскарадинг IPv4 для совместного использования интернет-соединения.
- Зона *dmz* — для общедоступных серверов в демилитаризованной зоне[1]. Принимаются только входящие SSH-соединения. (DMZ — это отдельный сегмент сети, предназначенный для серверов с выходом в Интернет.)
- Зона *work* — принимаются входящие запросы на соединение только для ssh и dhcpv6-client.

[1] https://ru.wikipedia.org/wiki/DMZ_(компьютерные_сети).

- Зона *home* — принимаются входящие запросы на соединение только для ssh, mdns samba-client и dhcpv6-client.
- Зона *internal* — принимаются входящие запросы на соединение только для ssh, mdns, samba-client и dhcpv6-client.
- Зона *trusted* — принимаются все входящие запросы на соединение.

Вы можете настроить любую из этих зон или создать новую (см. рецепт 14.9).

Дополнительная информация

- Рецепт 14.9.
- https://firewalld.org.

14.8. Изменение зоны firewalld по умолчанию

Задача

Зона, используемая по умолчанию, вас не устраивает, и вы хотите сменить ее.

Решение

Проверьте, какая зона используется по умолчанию:

```
$ firewall-cmd --get-default-zone
internal
```

Допустим, вы решили использовать зону *drop*, как наиболее ограничивающую. Сделать это можно с помощью команды `firewall-cmd`:

```
$ sudo firewall-cmd --set-default-zone drop
success
```

После выполнения этой команды не требуется перезагружать или перезапускать firewalld.

Комментарий

Зоны можно также назначать с помощью NetworkManager (см. рецепт 14.11). NetworkManager назначает зону по умолчанию для всех подключений, которым явно не назначена другая зона.

Дополнительная информация

- Подраздел «Комментарий» в рецепте 14.7, содержащий больше сведений о зонах firewalld.
- Рецепт 14.11.
- https://firewalld.org.

14.9. Настройка зон firewalld

Задача

Ни одна из предопределенных зон не соответствует вашим потребностям, и вы хотели бы изменить одну из них.

Решение

Предположим, что больше всего вам нравится зона *internal*, но ее настройки по умолчанию немного не соответствуют вашим нуждам. С текущими настройками эта зона пропускает запросы на соединение `ssh`, `mdns`, `samba-client` и `dhcpv6-client`:

```
$ firewall-cmd --zone=internal --list-all
internal
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: ssh mdns samba-client dhcpv6-client
[...]
```

Следующий пример показывает, как убрать поддержку `samba-client`, поскольку вы не используете Samba:

```
$ sudo firewall-cmd --remove-service=samba-client --zone=internal
success
```

Вы используете небольшой локальный сервер каталогов 389 Directory Server, и поэтому вам нужно добавить поддержку службы LDAPS:

```
$ sudo firewall-cmd --zone=internal --add-service=ldaps
success
```

Эти изменения носят временный характер и будут потеряны после перезапуска системы или перезагрузки конфигурации. Однако они вступают в действие

немедленно, что позволяет тут же протестировать их. Протестируйте свои изменения, и если все работает как надо, то сохраните их:

```
$ sudo firewall-cmd --runtime-to-permanent
success
```

Чтобы отменить изменения, выполните ту же самую команду, только вместо параметра `--runtime-to-permanent` используйте `--reload`. Этот параметр отменит временные изменения и вернет исходные настройки:

```
$ sudo firewall-cmd --reload
success
```

Комментарий

Параметр `--reload` не разрывает уже имеющихся активных соединений.

Параметр `--complete-reload` перезапускает firewalld и перезагружает модули ядра, вследствие чего имеющиеся активные соединения разрываются. Это хороший вариант для случаев, когда временные изменения настолько запутанны, что было бы желательно начать все сначала.

Дополнительная информация

- Подраздел «Комментарий» в рецепте 14.7, содержащий больше сведений о зонах firewalld.
- https://firewalld.org.
- Глава 4.

14.10. Создание новой зоны

Задача

Создать новую зону с уникальными настройками.

Решение

Создайте XML-файл с настройками зоны, затем перезагрузите брандмауэр firewalld, и он будет готов к использованию новой зоны.

Следующий пример создает зону для локальных служб имен с серверами DNS и DHCP на одном компьютере с доступом по SSH. Настройки сохраняются в файле `/etc/firewalld/zones/names.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<zone>
  <short>Name Services</short>
  <description>
    DNS and DHCP servers for the local network, IPv4 only.
  </description>
  <service name="dns"/>
  <service name="dhcp"/>
  <service name="ssh"/>
</zone>
```

Выполните команду `sudo firewall-cmd --get-zone`, и вы увидите, что ваша новая зона отсутствует в списке известных зон. Добавьте параметр `--permanent`, чтобы увидеть все новые зоны, не прочитанные брандмауэром firewalld, и теперь в списке появится зона *names*. Имена зон — это имена файлов без расширения `.xml`:

```
$ sudo firewall-cmd --permanent --get-zones
block dmz drop external home internal names public trusted work
```

Перезагрузите firewalld:

```
$ sudo firewall-cmd --reload
success
```

Теперь новая зона присутствует в общем списке, и firewalld готов прочитать ее настройки:

```
$ sudo firewall-cmd --get-zones
block dmz drop external home internal names public trusted work
```

И вывести их:

```
$ sudo firewall-cmd --zone=names --list-all
names
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: dhcp dns ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Новая зона готова к использованию, и ее можно изменить, как любую другую.

Комментарий

См. `man 5 firewalld.zone`, чтобы узнать больше о параметрах конфигурации, и используйте исходные файлы с настройками предопределенных зон в `/usr/lib/firewalld/zone/` в качестве примеров. Единственные файлы, которые попадают в `/etc/firewalld/zone/`, — это пользовательские файлы.

Чтобы удалить зону, удалите ее файл `.xml`, а затем перезагрузите firewalld.

Дополнительная информация

- `man 5 firewalld.zone`
- https://firewalld.org.
- Рецепт 14.9.

14.11. Интеграция NetworkManager и firewalld

Задача

Вам приходится подключать свой ноутбук к разным сетям: в офисах, кафе, отелях и в зонах для совместной работы. Поэтому вы хотели бы узнать, как настроить NetworkManager, чтобы учитывать особенности подключений и назначать им правильные зоны брандмауэра.

Решение

NetworkManager прекрасно интегрируется с firewalld. При подключении к новой сети NetworkManager назначает этому подключению зону firewalld по умолчанию.

В NetworkManager вы можете назначить определенному соединению другую зону, отличную от зоны по умолчанию. Щелкните на значке апплета NetworkManager на панели задач, чтобы открыть диалог Edit Connections (Настроить сетевые соединения) (рис. 14.2).

Или выполните команду `nm-connection-editor`, чтобы открыть редактор настроек. В диалоге выберите соединение, настройки которого хотите изменить, и нажмите кнопку со значком шестеренки. После этого откроется диалог, изображенный на рис. 14.3.

Откройте вкладку General (Основное) и в раскрывающемся списке Firewall Zone (Зона брандмауэра) выберите зону, которая должна быть назначена соединению. Сохраните изменения и закройте диалог.

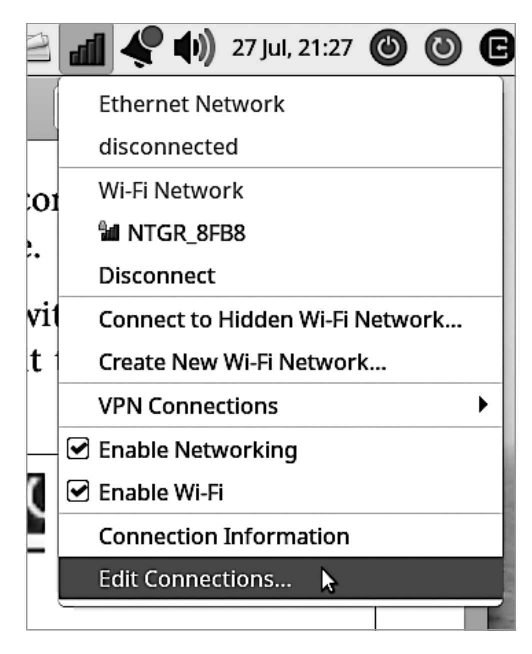


Рис. 14.2. Настройка сетевых соединений в NetworkManager

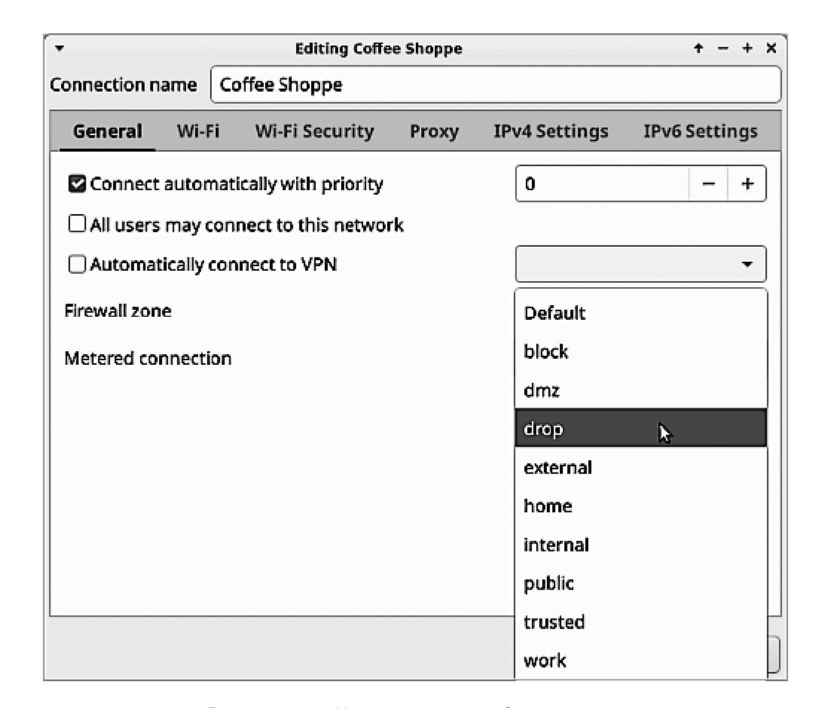


Рис. 14.3. Изменение зоны брандмауэра

Дополнительная информация

- Подраздел «Комментарий» в рецепте 14.7, содержащий больше сведений о зонах firewalld.
- Рецепт 14.9.
- Руководство по NetworkManager (https://oreil.ly/pvrwj).

14.12. Блокировка и разблокировка конкретных портов

Задача

Вы используете нестандартные порты, например порт 2022 для сервера SSH, и решили заблокировать порт 22 и открыть порт 2022.

Решение

Все зоны firewalld автоматически блокируют любые порты, которые явно не разрешены. Исключение составляет зона *trusted*, которая разрешает все порты. Если вы использовали службу SSH, которая по умолчанию использует TCP-порт 22, то сначала заблокируйте порт 22 в соответствующей зоне, а затем добавьте порт 2022 и перезагрузите firewalld. Ниже представлен пример назначения нестандартного порта в зоне *work*:

```
$ sudo firewall-cmd --zone=work --remove-port=22/tcp
success
$ sudo firewall-cmd --zone=work --add-port=2022/tcp
success
```

Проверьте настройки зоны:

```
$ sudo firewall-cmd --list-all --zone=work
work
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: ssh
  ports:2022/tcp
[...]
```

Добившись желаемого, сохраните свои изменения:

```
$ sudo firewall-cmd --runtime-to-permanent
```

Комментарий

Если при попытке заблокировать порт вы увидели сообщение Warning: NOT_ENABLED: 22:tcp (Внимание: не включен: 22:tcp), то это означает, что он не был разрешен в данной зоне. Просто игнорируйте это сообщение и добавьте новый порт.

При использовании нестандартных портов на сервере клиенты, подключающиеся к службе, должны указывать этот номер порта. Например, для SSH:

```
$ ssh -p 2022 server1
```

Как узнать, какие порты использовать? Каждая служба имеет свои порты по умолчанию, номера которых можно найти в документации с описанием службы и в файле `/etc/services`. Вы можете назначить нестандартные порты, которые не должны использоваться другими службами и должны находиться в диапазоне от 1024 до 49 151. Зафиксируйте свои изменения в `/etc/services`. Необходимо также настроить нестандартные порты в конфигурации сервера. См. пример в рецепте 12.3.

Дополнительная информация

- https://firewalld.org.
- Рецепт 12.3.

14.13. Блокировка IP-адресов с помощью своих правил

Задача

Заблокировать определенные IP-адреса.

Решение

Создайте дополнительное правило, определяющее блокируемые адреса и действие (например, действие `reject`, как в следующем примере). Ниже представлен пример блокировки одного адреса в зоне *internal*:

```
$ sudo firewall-cmd --zone=internal \
  --add-rich-rule='rule family="ipv4" source address=192.168.1.91 reject'
success
```

Протестируйте правило, выполнив эхо-запрос с заблокированного хоста. На заблокированном хосте должно появиться сообщение Destination Port Unreachable (Порт назначения недоступен).

Если вы решили не сохранять правило, то выполните команду `sudo firewall-cmd --reload`, чтобы удалить его.

Чтобы сохранить правило, выполните команду с параметром `--runtime-to-permanent`:

```
$ sudo firewall-cmd --runtime-to-permanent
```

Выведите список дополнительных правил в зоне:

```
$ sudo firewall-cmd --zone=internal --list-rich-rules
rule family='ipv4' source address='192.168.1.91' reject
```

Чтобы удалить сохраненное правило, используйте параметр `--remove-rich-rule`:

```
$ sudo firewall-cmd --zone=internal \
        --remove-rich-rule="rule family='ipv4' \
        source address='192.168.1.91' reject"
success
```

Если не требуется полностью блокировать какой-то хост, то для него можно заблокировать доступ лишь к определенным службам. Ниже представлен пример блокировки доступа только к службе SSH:

```
$ sudo firewall-cmd --zone=internal --add-rich-rule='rule family="ipv4" \
 source address=192.168.1.91 service name="ssh" protocol=tcp reject'
success
```

Комментарий

В зоне можно создать несколько сложных правил, но будьте осторожны, чтобы избежать конфликтов.

Однажды человек, с которым я имела сомнительное удовольствие работать вместе, подумал, что забавно было бы попрактиковать на своих коллегах навыки проникновения. В нашей команде было несколько тестовых серверов на наших рабочих станциях, доступных всем членам команды. Наш горе-взломщик настолько достал всех своими попытками взлома, что мы все заблокировали его в настройках наших брандмауэров.

Дополнительная информация

- Подраздел «Комментарий» в рецепте 14.7, содержащий больше сведений о зонах и параметрах firewalld.
- https://firewalld.org.
- `man 5 firewalld.richlanguage`

14.14. Изменение действия по умолчанию для зоны

Задача

Изменить действие, выполняемое зоной по умолчанию.

Решение

Проверьте текущее действие по умолчанию:

```
$ sudo firewall-cmd --zone=internal --list-all
internal
  target: ACCEPT
[...]
```

Замените `ACCEPT` на `REJECT`, перезагрузите firewalld и проверьте:

```
$ sudo firewall-cmd --permanent --zone=internal --set-target=REJECT
success

$ sudo firewall-cmd --reload

$ firewall-cmd --zone=names --list-all
names
  target: REJECT
[...]
```

Комментарий

Действие по умолчанию определяет, что должна делать зона с пакетами, которые не подошли ни под одно правило. Может иметь одно из четырех значений: `default`, `ACCEPT`, `DROP` или `REJECT`.

Дополнительная информация

- https://firewalld.org.
- Подраздел «Комментарий» в рецепте 14.5, содержащий больше сведений о зонах firewalld.
- Рецепт 14.11.

ГЛАВА 15

Печать в Linux

Для управления принтерами в Linux используется система печати для Unix CUPS (Common Unix Printing System). В этой главе вы узнаете, как устанавливать драйверы и управлять принтерами, а также как открывать доступ к ним по сети. Вы узнаете о будущем печати *без драйверов* в Linux, когда принтеры будут доступны клиентам без установки драйверов.

Обзор

Ключ к беспроблемной печати в Linux — выбор качественных принтеров и многофункциональных устройств (совмещающих в себе принтер, сканер, копир, факс), которые хорошо поддерживаются в Linux. К счастью, это намного проще, чем в былые времена. Выбрав хорошо поддерживаемое устройство, драйверы для которого включены в состав CUPS, вам не придется беспокоиться о самостоятельном поиске и скачивании драйверов.

Следующий хороший вариант — покупка устройств, поставляемых вместе с драйверами для Linux. Мне такой вариант не особенно нравится, поскольку драйверы часто устаревшие и не поддерживаются производителем, а кроме того, их приходится устанавливать вручную. Это особенно характерно для многофункциональных устройств (МФУ), например Brother MFC-J5945DW, которым я пользуюсь. Для данного устройства нет собственных драйверов в CUPS, хотя оно поддерживает печать без драйверов. Для меня это была выгодная покупка, и чернила стоят недорого, хотя сейчас, получив определенный опыт, я купила бы устройство со встроенной поддержкой Linux. Она более надежна, поскольку, если устройство поддерживается в CUPS, оно всегда будет поддерживаться и вы не попадете в зависимость от производителя, не поддерживающего свои драйверы или прекращающего скачивание драйверов.

Наименее предпочтительный вариант — покупка в надежде, что все заработает, невзирая на прошлый опыт. В таком случае для принтеров, не поддерживаемых

в Linux, можно попробовать использовать драйверы macOS (файлы PPD), хотя это может потребовать усилий с вашей стороны, если файл PPD содержит некоторые записи, специфичные для macOC, такие как вызов выполняемых файлов, библиотек или фильтров macOS. Их необходимо заменить эквивалентами в Linux, если они существуют. См. раздел cupsFilter (https://oreil.ly/w3Oqd), где приводится информация, которая может пригодиться вам, если вы решитесь на этот шаг.

Если вам понадобится использовать устройство совместно с коллегами, то наименее хлопотный вариант — приобрести устройство со встроенной поддержкой сети и встроенными средствами управления копированием, настройки сети, просмотра уровней чернил, очистки печатающих головок и выполнения других задач по настройке и обслуживанию. Такие устройства намного проще и удобнее в эксплуатации, чем устройства, требующие настройки и управления с компьютера.

Поиск поддерживаемых принтеров и сканеров

Принтеры и многофункциональные устройства Hewlett-Packard (HP) имеют отличную поддержку Linux в виде пакетов *hplip*, *hplip-hpijs*, *hplip-sane* и *hplip-scan-utils*. Конечно, каждый дистрибутив Linux имеет свои особенности и присваивает пакетам разные имена, такие как *hpijs-ppds*, *hplip-data*, *printer-driver-hpcups*, *hplip-common* и *libsane-hpaio*. Однако поиск по строке *hplip* должен помочь найти их.

Не все принтеры и МФУ компании Hewlett-Packard поддерживаются в Linux; см. ссылку ниже на базу данных со списком устройств HP, поддерживаемых в Linux.

Компания Brother предлагает хорошие устройства, достойную поддержку клиентов и привлекательные цены на чернила. У них есть и устройства со встроенной поддержкой Linux, и устройства, для которых нужны драйверы Brother.

Canon, Epson, Honeywell, Fujitsu, IBM, Lexmark, Kodak, Tektronix, Samsung, Sharp, Xerox, Toshiba и многие другие бренды предлагают определенный уровень поддержки Linux. Иногда может быть сложно узнать, какие модели поддерживаются. Некоторые производители сообщают об этом в спецификациях своих продуктов. Есть несколько сайтов, которые стоит проверить, хотя они, как правило, содержат неполную информацию и редко обновляются, но они вполне могут послужить отправной точкой:

- поддерживаемые принтеры HP (https://oreil.ly/y9z4J);
- списки принтеров OpenPrinting.org (https://oreil.ly/7JbPH);
- принтеры и МФУ на сайте H-node (https://oreil.ly/Hwy0w);

- магазин ThinkPenguin (https://oreil.ly/54H5F);
- страница Ubuntu со списком поддерживаемых принтеров (https://oreil.ly/03SV3);
- принтеры IPP Everywhere (https://oreil.ly/l7pFz).

Драйверы для принтеров в CUPS

Драйверы принтеров для Linux распространяются в составе системы печати для Unix CUPS (Common Unix Printing System). CUPS — это подсистема печати, ставшая стандартом для Linux примерно в 2000 году. Apple начала использовать CUPS примерно в 2002 году, затем наняла создателя CUPS Майкла Свита (Michael Sweet) и купила исходный код в 2007 году. Свит покинул Apple в 2019 году, и работа над CUPC в Apple остановилась; см. страницу apple/cups на GitHub (https://oreil.ly/HgUX8). Однако мистер Свит не сидел сложа руки; он усердно работал над производной версией OpenPrinting.org CUPS в OpenPrinting/cups на GitHub (https://oreil.ly/uP0CJ).

CUPS — это гораздо больше, чем просто программное обеспечение. Майкл Свит, Тилль Кампетер (Till Kamppeter) и другие приложили немало усилий, чтобы привлечь производителей и разработать общие стандарты печати и API. Центрами разработки CUPS и стандартов печати стали The Printer Working Group (https://oreil.ly/yEMad) и OpenPrinting (https://oreil.ly/caH6b).

Драйверы принтера в CUPS состоят из одного или нескольких фильтров для конкретного принтера, упакованных в файлы PPD (PostScript Printer Description — описание принтера PostScript). Для работы с любым принтером в CUPS, даже не поддерживающим PostScript, нужен файл PPD. PPD-файлы содержат описания принтеров, их команд и фильтры.

Фильтры преобразуют задания на печать в формат, понятный принтеру, например PDF, HPPCL, растр и изображение, и передают команды для таких операций, как выбор страницы, размер бумаги, цвет, контраст и тип носителя. PPD — это простые текстовые файлы, и по умолчанию файлы PPD для всех поддерживаемых принтеров хранятся в каталоге `/usr/share/cups/model/`. Файлы PPD для установленных принтеров хранятся в `/etc/cups/ppd/`.

Время PPD уходит

Система печати CUPS с самого начала полагалась на файлы PPD, и они хорошо зарекомендовали себя. Однако появился новый подход, называемый *печатью без драйверов*. Вместо использования статических файлов PPD принтер сам сообщает о своих возможностях и не требует установки

драйверов на клиентских машинах. Идея состоит в том, чтобы сделать подключение к новому принтеру таким же простым, как подключение к новой сети с помощью NetworkManager, который автоматически находит доступные сети и не требует установки драйверов или настройки каждой новой сети вручную. Это особенно удобно для мобильных устройств, таких как телефоны и планшеты, имеющих ограниченный объем памяти для хранения драйверов и ограниченный размер экрана.

Бездрайверная технология была представлена в CUPS 2.2.0. Но для большей надежности желательно установить версию CUPS 2.2.4 (выпущенную в июне 2017 года) или выше. Узнать больше можно на следующих ресурсах:

- приложения для печати: новый подход к печати в Linux (https://oreil.ly/clNC3);
- печать без драйверов в CUPS (https://oreil.ly/yaj5q).

15.1. Использование веб-интерфейса CUPS

Задача

Найти инструмент администрирования CUPS.

Решение

Откройте веб-интерфейс CUPS в своем браузере, доступный по адресу http://localhost:631/ (рис. 15.1).

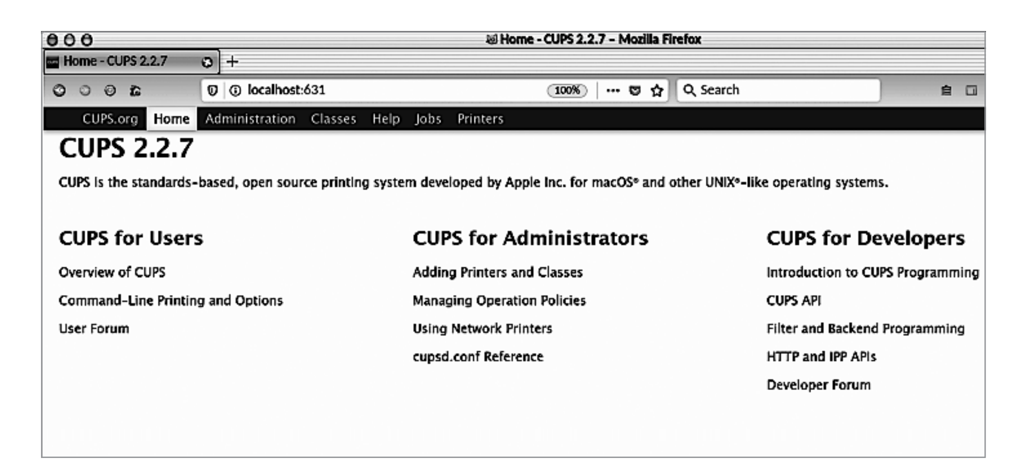


Рис. 15.1. Веб-интерфейс CUPS

Комментарий

Существует множество графических инструментов для управления принтерами, например *system-config-printer* и модуль управления принтерами для YaST в openSUSE. Однако веб-интерфейс CUPS предоставляет наиболее полные возможности управления и одинаков для всех дистрибутивов Linux.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.2. Установка принтера, подключенного непосредственно к компьютеру

Задача

Установить новый принтер, подключенный непосредственно к вашему компьютеру. Предполагается, что вы, поступив мудро, приобрели принтер со встроенной поддержкой в CUPS.

Решение

Откройте веб-интерфейс CUPS. Ваш принтер должен быть подключен к компьютеру и включен. Далее демонстрируются примеры настройки в системе Linux Mint.

Перейдите на вкладку Administration (Администрирование) и нажмите кнопку Add Printer (Добавить принтер). Вам будет предложено ввести имя пользователя и пароль (рис. 15.2). (Если вам не удается выполнить вход со своей учетной записью, но работает вход с учетной записью root, то см. рецепт 15.7, чтобы узнать, как настроить CUPS для входа без полномочий root.) Установите флажок Save debugging information for troubleshooting (Сохранять отладочную информацию в журнале) и флажок Share printers connected to this system (Разрешить совместный доступ к принтерам, подключенным к этой системе), чтобы разрешить общий доступ к принтерам, напрямую подключенным к вашему компьютеру. Так вы только включите поддержку общего доступа, но далее вам придется включить общий доступ для каждого принтера, который вы решите предоставить в общий доступ.

На следующем экране CUPS отыщет и перечислит ваши принтеры в разделе Local Printers (Локальные принтеры) (рис. 15.3). Выберите свой принтер и нажмите кнопку Continue (Продолжить).

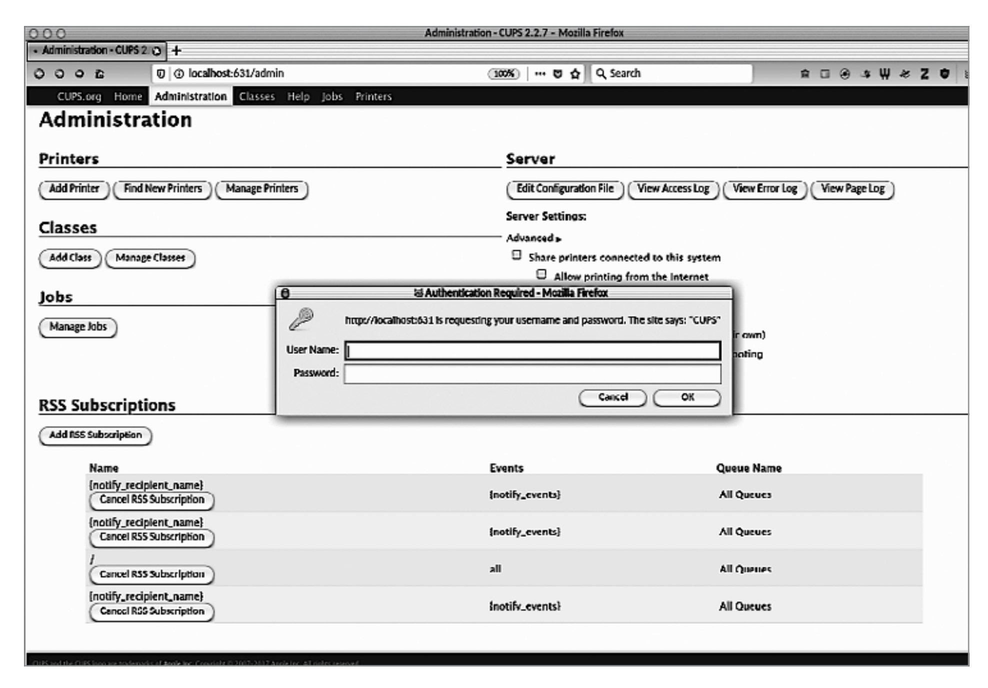


Рис. 15.2. Добавление принтера

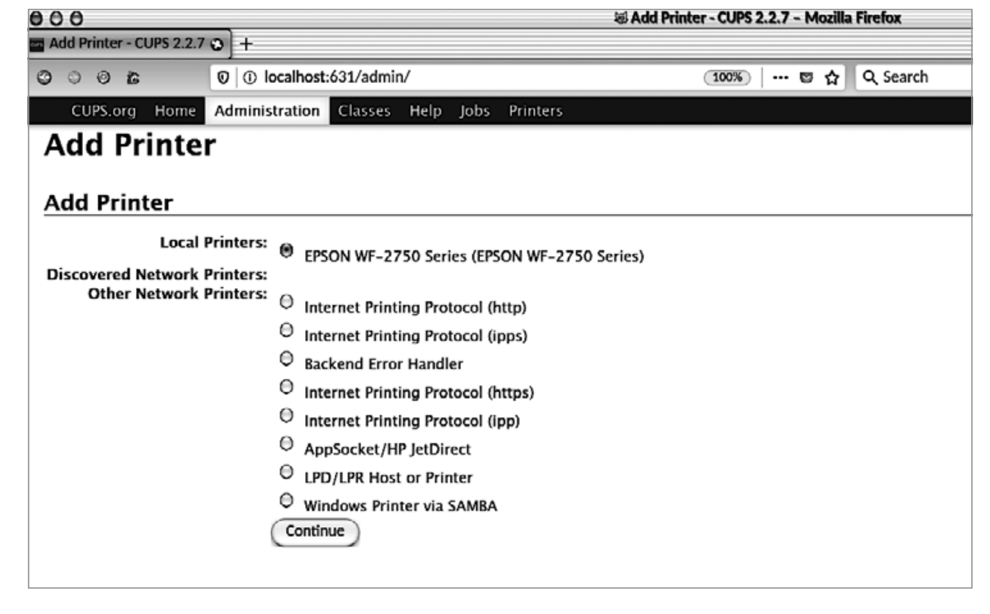


Рис. 15.3. CUPS отыщет ваш локальный принтер

Теперь вы должны увидеть страницу, как показано на рис. 15.4, с полями Name (Название), Description (Описание) и Location (Местоположение). Поля Name (Название) и Description (Описание) заполняются автоматически, но вы можете изменить их, как вам заблагорассудится. Поле Name (Название) будет отображаться в диалоговых окнах выбора принтера при печати документа.

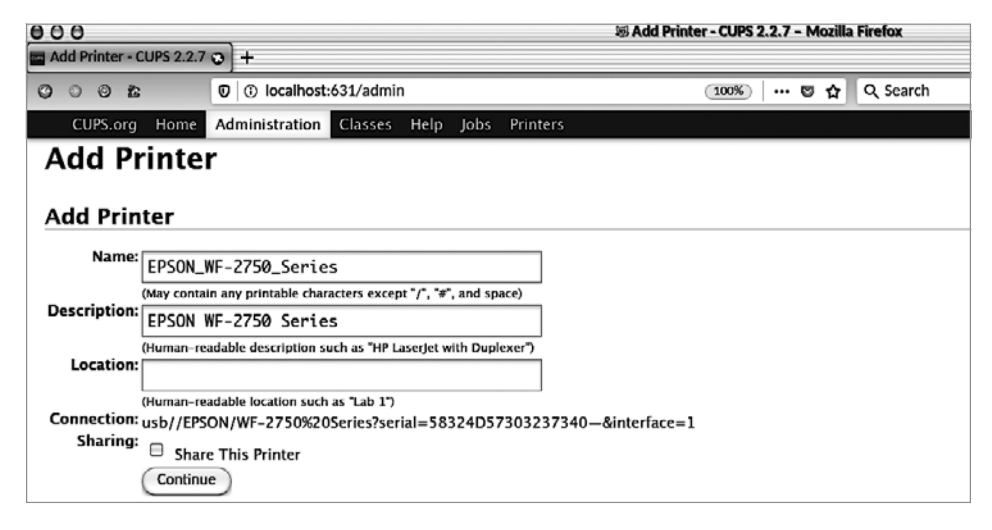


Рис. 15.4. Укажите название, описание и местоположение

Выберите драйвер принтера. CUPS выведет огромный список драйверов, доступных для выбора. Найдите драйвер для своей модели принтера. Драйвер, выбранный на рис. 15.5, был установлен в составе пакета *epson-inkjet-printer-escpr* (в Ubuntu соответствующий пакет называется *printer-driver-escpr*), он предназначен для цветных струйных принтеров Seiko Epson.

Заключительная страница настройки позволяет изменить параметры по умолчанию, такие как тип и размер бумаги, цветная или черно-белая печать, качество печати и т. д., в зависимости от того, что поддерживает ваш принтер и его драйвер. Закончив настройки, нажмите кнопку Set Default Options (Сохранить параметры) (рис. 15.6).

После этого вы увидите страницу Printers (Принтеры) со списком установленных принтеров (рис. 15.7).

Щелкните на вновь установленном принтере и напечатайте пробную страницу, выбрав в раскрывающемся списке Maintenance (Обслуживание) пункт Print a test page (Печать пробной страницы). Если страница была благополучно напечатана, значит, вы все сделали правильно.

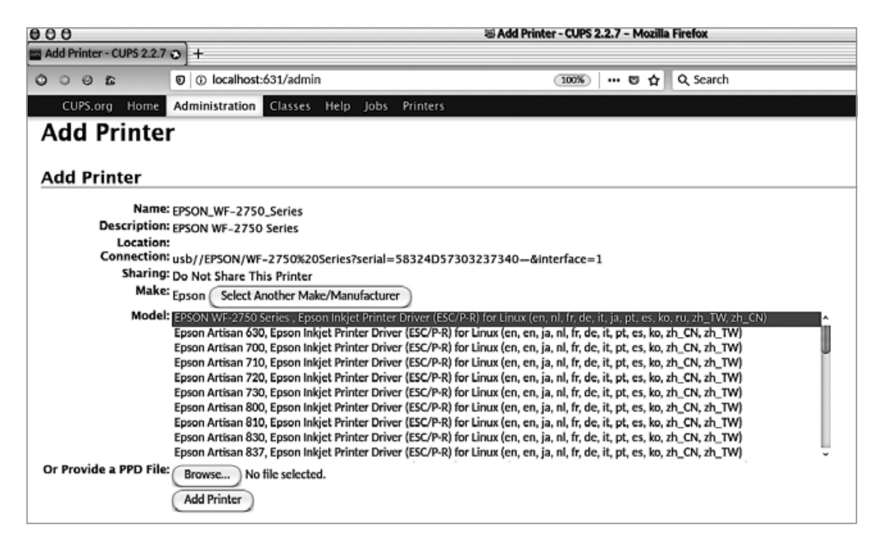


Рис. 15.5. Выберите драйвер принтера

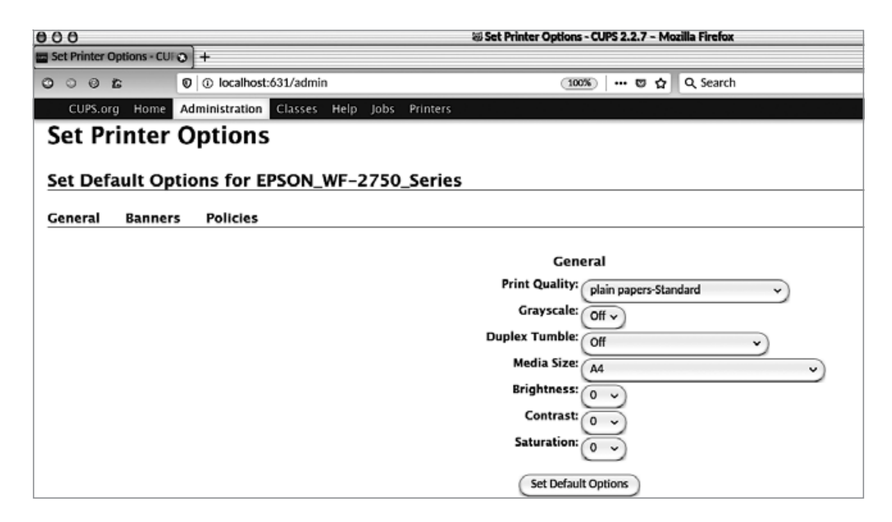


Рис. 15.6. Настройка параметров печати по умолчанию

Printers - CUPS 2.2.7 - Mozilla Firefox

localhost:631/printers/

CUPS.org Home Administration Classes Help Jobs Printers

Printers

Search in Printers: Search Clear

Showing 1 of 1 printer.

Queue Name	Description	Location	Make and Model	Status
EPSON_WF-2750_Series	EPSON WF-2750 Series		EPSON WF-2750 Series , Epson Inkjet Printer Driver (ESC/P-R) for Linux	Idle

Рис. 15.7. Все установленные принтеры

Комментарий

На рис. 15.2 можно заметить две кнопки: Add Printer (Добавить принтер) и Find New Printers (Найти новый принтер). Это практически одно и то же, просто списки обнаруженных принтеров организованы по-разному.

Вероятно, вам на выбор будет предложено несколько драйверов принтера; например, часто для одного и того же принтера предлагаются драйверы CUPS+Gutenprint и Foomatic. Раньше Gutenprint был лучшим выбором для цветных принтеров, но я советую попробовать оба и выбрать тот, который больше понравится. Драйверы CUPS+Gutenprint Simplified предлагают меньше возможностей и параметров, чем полные версии.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.3. Выбор имен для принтеров

Задача

В диалоге печати документа вам предлагается выбрать из нескольких принтеров, порой очень похожих по внешнему виду, и вы затрудняетесь с выбором.

Решение

При установке принтера введите описательное имя в поле Name (Название) (рис. 15.8). Это обязательно делать при установке, поскольку потом изменить название принтера нельзя.

Комментарий

При установке принтера с помощью веб-интерфейса CUPS можно также изменить поля Description (Описание) и Location (Местоположение). Но многие приложения, поддерживающие печать, читают только поле Name (Название). В числе исключений можно назвать почтовый клиент Evolution и браузеры Firefox и Chromium, которые отображают название, местоположение и статус.

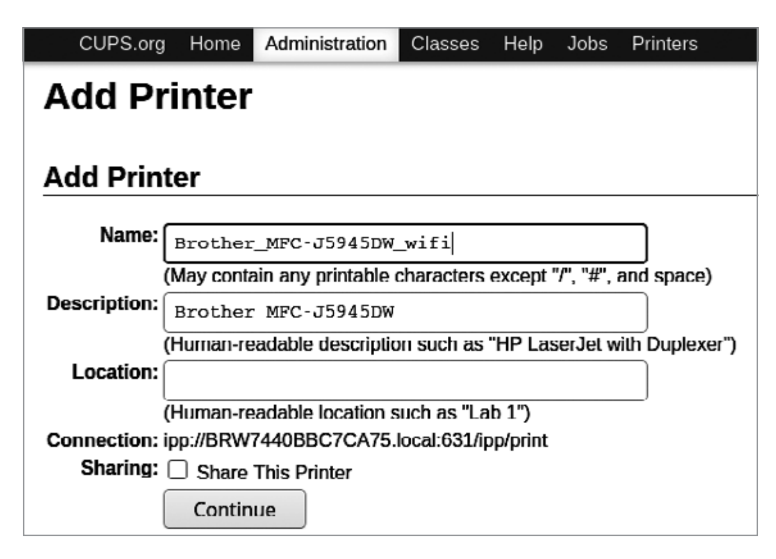


Рис. 15.8. Выбирайте описательные имена для принтеров при установке

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.4. Установка сетевого принтера

Задача

В вашей сети есть общий сетевой принтер, и вы хотите получить возможность печатать на нем со своего компьютера.

Решение

Процедура установки сетевого принтера аналогична установке принтера, подключенного к порту USB (см. рецепт 15.2), с той лишь разницей, что вы должны выбрать обнаруженный сетевой принтер. Принтер должен быть включен и находиться в том же сегменте сети, что и ваш компьютер. Вы увидите его в списке Discovered Network Printers (Найденные сетевые принтеры) (рис. 15.9).

При этом на клиентах вы должны открыть TCP-порт с номером 631 в настройках брандмауэра.

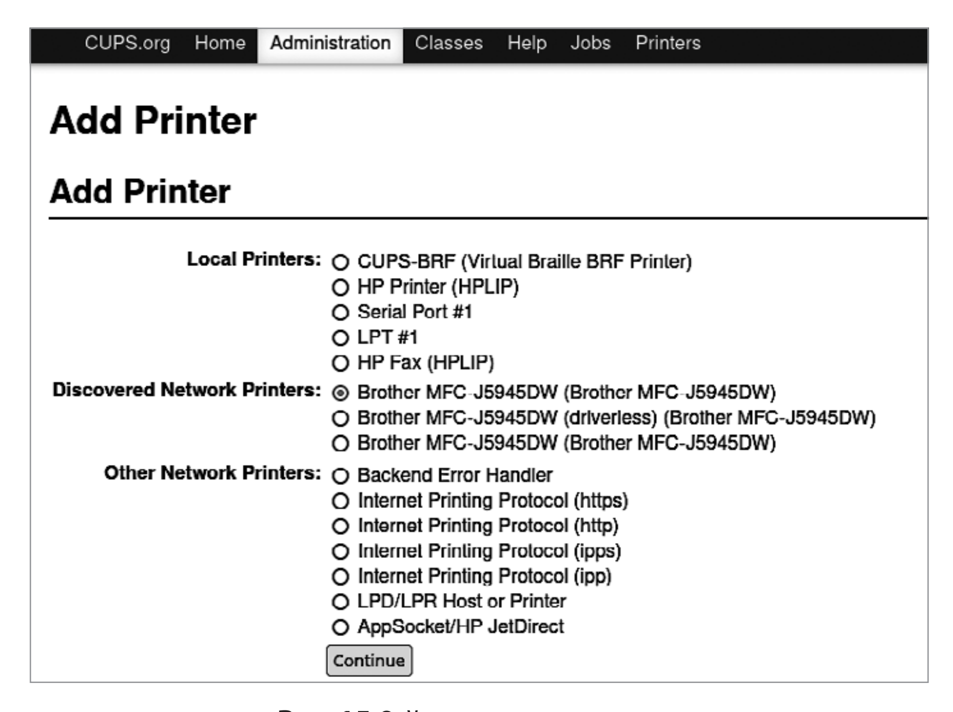


Рис. 15.9. Установка сетевого принтера

Комментарий

А что если CUPS не обнаруживает принтер? См. рецепт 15.11, где немного рассказывается о поиске и устранении неисправностей. Если CUPS не видит ваш принтер, то вы не сможете его установить.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.5. Печать без драйверов

Задача

Ваш принтер не поддерживается системой печати CUPS, и вы хотите попробовать настроить печать без драйверов. Или, может быть, вы задумали подключить принтер к своему устройству на Android или iOS.

Решение

Возможно, вы уже видели варианты driverless (без драйверов) в списке выбора драйвера принтера CUPS. В следующем примере показаны шаги по настройке моего принтера Brother MFC-J5945DW, для которого нет встроенной поддержки CUPS.

Перейдите на страницу Administration ▸ Add Printer (Администрирование ▸ Добавить принтер) в веб-интерфейсе CUPS. Система печати CUPS обнаружила мой принтер Brother и отобразила его в списке Discovered Network Printers (Найденные сетевые принтеры) (рис. 15.10). В списке присутствует вариант driverless, и это правильный выбор.

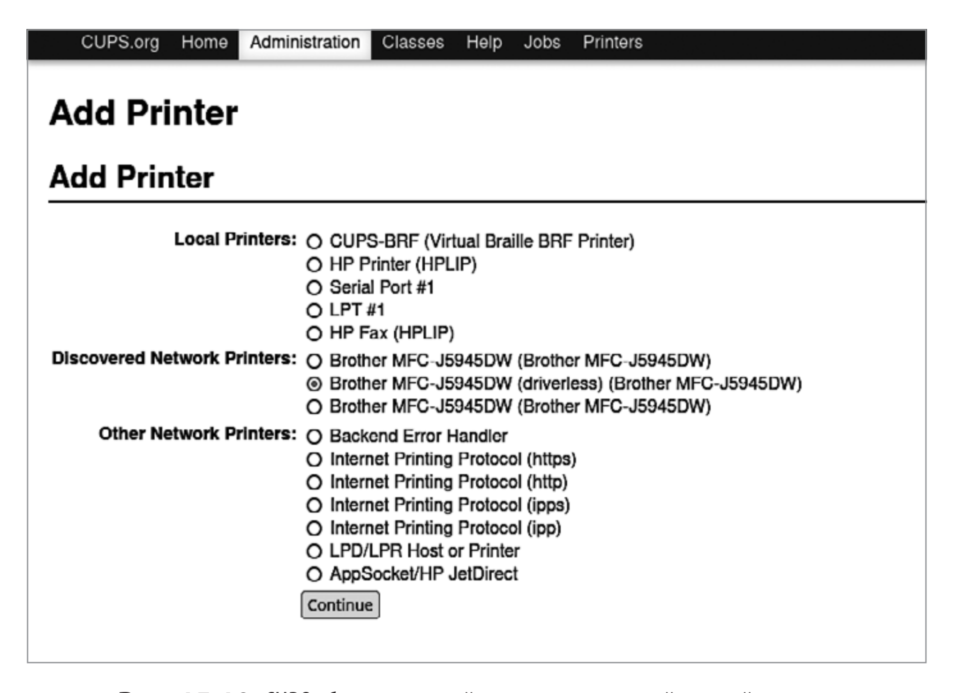


Рис. 15.10. CUPS обнаружила мой неподдерживаемый сетевой принтер

Продолжите установку и выберите правильный драйвер, которым в моем случае (как показано на рис. 15.11) является Brother MFC-J5945DW, driverless, cups-filters 1.25.0 (en).

Напечатайте пробную страницу, и если она выглядит правильно, то установку можно считать завершенной.

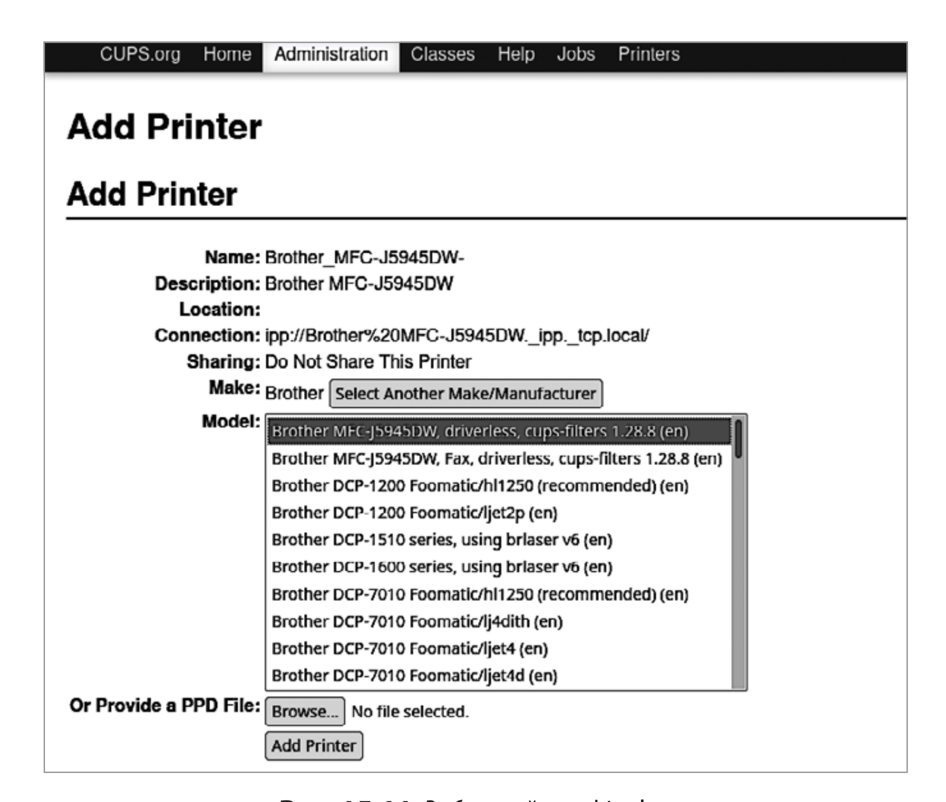


Рис. 15.11. Выбор драйвера driverless

Комментарий

Строго говоря, это не печать без драйверов, поскольку в действительности CUPS создает файлы PPD в `/etc/cups/ppd` для вашего «бездрайверного» принтера. Однако вам не нужно поддерживать каталоги, заполненные файлами PPD от OpenPrinting.org и Gutenprint.

Ваш принтер должен поддерживать печать без драйверов, а это значит, что он должен поддерживать стандарты Mopria, AirPrint, IPP Everywhere или WiFi Direct Print. Все они похожи между собой: принтер автоматически сообщает о своих возможностях и свой сетевой адрес с помощью демона Avahi. Avahi обеспечивает обнаружение служб в локальной сети, используя набор протоколов mDNS/DNS-SD. (Apple называет эту службу Bonjour и Zeroconf.)

Печать без драйверов в CUPS отлично сочетается с устройствами на Android и iOS. Вам нужно только установить приложение для принтера. Если ваш принтер поддерживает печать без драйвера, и особенно если он сертифицирован

Mopria, то вашим мобильным устройствам не составит труда его найти. Сертификация в Mopria означает, что принтер поддерживает беспроводную печать с мобильных устройств. Если в документации к вашему принтеру не указано, что он сертифицирован в Mopria, то выполните следующую команду, которая покажет, сертифицирован ли принтер:

```
$ avahi-browse -rt _ipp._tcp
[...]
txt = ["mopria-certified=1.3"
[...]
```

Дополнительная информация

- Вики-страница Debian под названием Driverless Printing (https://oreil.ly/d2Qw8).
- Документация для CUPS (https://oreil.ly/OlCzV).

15.6. Совместное использование несетевых принтеров

Задача

Организовать совместное использование принтера, не имеющего встроенной поддержки подключения к сети.

Решение

CUPS обеспечивает возможность совместного доступа к принтерам, не подключенным к сети, но подключенным к персональным компьютерам в вашей сети. Прежде всего у вас должна действовать служба имен, чтобы все ваши локальные узлы могли обнаруживать друг друга.

Включите общий доступ к принтеру на странице Administration (Администрирование), установив флажок Share printers connected to this system (Разрешить совместный доступ к принтерам, подключенным к этой системе). Затем включите общий доступ к конкретному принтеру (рис. 15.12).

CUPS объявит о присутствии принтера в вашей сети. Любой клиент Linux в вашей сети, который захочет использовать этот принтер, должен установить его так же, как сетевой или локальный принтер; начните со страницы Administration ▸ Add Printer (Администрирование ▸ Добавить принтер), затем пройдите остальную часть процесса установки.

Рис. 15.12. Включение общего доступа к принтеру

Общие принтеры также могут использовать клиенты Windows и macOS. Операционная система macOS поддерживает обнаружение через DNS-SD/mDNS и IPP. Поддержка DNS-SD/mDNS обеспечивается демоном Avahi в Linux и называется Bonjour в Macintosh. Используйте панель управления в Macintosh, чтобы найти и установить доступные принтеры CUPS.

Windows 10 имеет встроенную поддержку DNS-SD/mDNS. Старые версии Windows поддерживают доступ к сетевым принтерам по протоколу интернет-печати (Internet Printing Protocol, IPP). Используйте панель управления принтерами в Windows, чтобы найти и установить общие принтеры CUPS.

Комментарий

В прежние времена, еще до того, как сетевые принтеры стали обычным явлением, администраторы использовали выделенные серверы печати. Это были старые персональные компьютеры, ноутбуки, небольшие одноплатные компьютеры или коммерческие серверы печати. Вы тоже можете купить небольшие устройства для организации серверов печати, которые стоят намного дешевле, чем в старые времена.

В настоящее время большинство принтеров имеют встроенную поддержку сети и ими проще управлять.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.7. Исправление сообщения об ошибке Forbidden

Задача

При попытке выполнить какую-либо административную задачу в веб-интерфейсе CUPS, например добавить новый принтер, вход с вашими учетными данными не выполняется и появляется сообщение Add Printer Error Unable to add printer: Forbidden (Ошибка добавления принтера. Невозможно добавить принтер: запрещено).

Решение

В некоторых дистрибутивах Linux, таких как openSUSE, конфигурация по умолчанию позволяет выполнять задачи администрирования CUPS только пользователю root. Отредактируйте файл `/etc/cups/cups-files.conf`, разрешив пользователям без полномочий root выполнять задачи администрирования CUPS. Найдите следующие строки:

```
# Administrator user group, used to match @SYSTEM in cupsd.conf policy rules...
# This cannot contain the Group value for security reasons...
SystemGroup root
```

Вот почему разрешается вход только с учетными данными root. Добавьте сюда свою личную группу, как в следующем примере, где была добавлена группа `duchess` — личная группа пользователя Duchess:

```
SystemGroup root duchess
```

Сохранив изменения в файле `/etc/cups/cups-files.conf`, перезапустите службу CUPS:

```
$ sudo systemctl restart cups.service
```

Теперь пользователь Duchess сможет выполнять административные задачи по управлению системой печати CUPS.

Другое решение — добавить системную группу, созданную специально для данной цели. В дистрибутивах Ubuntu Linux эта группа называется `lpadmin`, а в Fedora — группы `sys` и `wheel`. Вы можете создать свою группу для администрирования CUPS, как в следующем примере, где создается группа `cupsadmin` и в нее добавляется пользователь Mad Max:

```
$ sudo groupadd -r cupsadmin
$ sudo usermod -aG cupsadmin madmax
```

Mad Max должен выйти и снова войти в систему, чтобы активировать свое членство в группе. Добавьте группу `cupsadmin` в параметр `SystemGroup` в файле `/etc/cups/cups-files.conf`:

```
SystemGroup root duchess cupsadmin
```

Перезапустите CUPS, и после этого Mad Max сможет приступить к работе.

Комментарий

В том же файле `/etc/cups/cups-files.conf` должен иметься такой раздел:

```
# Default user and group for filters/backends/helper programs; this cannot be
# any user or group that resolves to ID 0 for security reasons...
#User lp
#Group lp
```

Ни одна из групп, перечисленных в `SystemGroup`, не может указываться в `Group`. Если вы попытаетесь использовать `lp`, как в предыдущем примере, то CUPS просто не запустится и выведет сообщения об ошибках в `/var/log/cups/error_log` или в системный журнал, в зависимости от настроек файла `/etc/cups/cups-files.conf`.

Если ваш дистрибутив использует систему инициализации SysV Init вместо systemctl, то перезапускайте CUPS с помощью такой команды:

```
$ sudo /etc/init.d/cups restart
```

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).
- Глава 4.

15.8. Установка драйверов принтеров

Задача

Узнать, насколько полный набор драйверов принтеров устанавливается с CUPS и есть ли другие драйверы, которые могут понадобиться, но не включены в CUPS.

Решение

Большинство дистрибутивов Linux устанавливают лишь некоторое подмножество доступных вариантов для печати. Каждый дистрибутив включает свой

набор драйверов принтеров в установку по умолчанию и определяет конкретные имена для пакетов, что особенно характерно для Ubuntu.

Ниже перечислен базовый набор пакетов CUPS и драйверов принтера:

- *cups* (сервер и клиент);
- *cups-filters* (фильтры и утилиты OpenPrinting CUPS);
- *gutenprint* (драйверы принтеров Gutenprint);
- *foomatic* (драйверы принтеров Foomatic);
- файлы PPD от OpenPrinting.org; например, OpenSUSE предлагает:
  - *OpenPrintingPPDs*;
  - *OpenPrintingPPDs-ghostscript* (интерпретатор драйверов принтеров, написанных на языке PostScript language);
  - *OpenPrintingPPDs-hpijs* (поддержка принтеров HP);
  - *OpenPrintingPPDs-postscript*;
- *cups-client* (утилиты командной строки для настройки и управления принтерами).

OpenPrinting.org включает Foomatic. Fedora и Ubuntu поставляют пакеты *foomatic*, а OpenSUSE включает пакеты *OpenPrinting*. Имена разные, но суть та же.

Этих пакетов может быть вполне достаточно. Тем не менее вот еще несколько пакетов с поддержкой печати, которые могут оказаться полезными:

- *gimp-gutenprint* (предоставляет более функциональный диалог печати для графического редактора GIMP (GNU Image Manipulation Program);
- *bluez-cups* (поддержка подключения принтеров по Bluetooth);
- *cups-airprint* (поддержка совместного использования принтеров с устройствами iOS);
- *ptouch-driver* (поддержка принтеров этикеток серии Brother P-touch);
- *rasterview* (средство для просмотра растровых изображений Apple, таких как GIF, JPEG и PNG, см. MSweet.org/rasterview (https://oreil.ly/zZZAp));
- *c2esp* (поддержка некоторых многофункциональных устройств Kodak).

Ubuntu содержит самую большую коллекцию драйверов для принтеров. Многие, но не все, имена пакетов начинаются с *printer-driver*:

- *openprinting-ppds* (поддержка печати OpenPrinting, файлы PostScript PPD);
- *printer-driver-all* (метапакет с драйверами для принтеров);
- *printer-driver-brlaser* (поддержка некоторых лазерных принтеров Brother);

- *printer-driver-c2050* (драйвер для струйного цветного принтера Lexmark 2050 Color Jetprinter);
- *printer-driver-foo2zjs* (драйвер для принтеров на основе ZjStream);
- *printer-driver-c2esp* (драйвер для семейства цветных струйных принтеров Kodak ESP AiO);
- *printer-driver-cjet* (драйвер для лазерных принтеров Canon LBP);
- *printer-driver-cups-pdf* (печать в файлы PDF через CUPS);
- *printer-driver-dymo* (драйвер для принтеров этикеток DYMO);
- *printer-driver-escpr* (драйверы для принтеров Epson Inkjets, использующих ESC/P-R);
- *printer-driver-fujixerox* (драйвер для принтеров Fuji Xerox);
- *printer-driver-gutenprint* (драйверы принтеров для CUPS);
- *printer-driver-hpcups* (HP Linux Printing and Imaging, драйвер CUPS Raster (hpcups));
- *printer-driver-hpijs* (HP Linux Printing and Imaging, драйвер для принтера (hpijs));
- *printer-driver-indexbraille* (печать через CUPS на принтерах Index Braille);
- *printer-driver-m2300w* (драйверы для цветных лазерных принтеров Minolta magicolor 2300W/2400W);
- *printer-driver-min12xxw* (драйвер для KonicaMinolta PagePro 1[234]xxW);
- *printer-driver-oki* (драйверы для принтеров OKI Data);
- *printer-driver-pnm2ppa* (драйвер для принтеров HP-GDI);
- *printer-driver-postscript-hp* (описание PostScript-принтеров HP);
- *printer-driver-ptouch* (драйверы для принтеров этикеток Brother P-touch);
- *printer-driver-pxljr* (драйверы для принтеров HP Color LaserJet 35xx/36xx);
- *printer-driver-sag-gdi* (драйверы для принтеров Ricoh Aficio SP 1000s/SP 1100s);
- *printer-driver-splix* (драйверы для принтеров Samsung и Xerox SPL2 и SPLc la).

Комментарий

Если вы не увидели свой принтер в списке драйверов в веб-интерфейсе CUPS, то попробуйте поискать по названию бренда вашего принтера в диспетчере пакетов. Все это может показаться довольно запутанным, но, к сожалению, настройка принтеров далека от идеала на всех вычислительных платформах (не только в Linux).

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).
- Документация для Ghostscript (https://oreil.ly/CHZpP).
- OpenPrinting (https://oreil.ly/jpYIW).
- The Printer Working Group (https://oreil.ly/Q5BUh).

15.9. Изменение настроек установленного принтера

Задача

Изменить настройки уже установленного принтера. Например, вы решили открыть к нему совместный доступ.

Решение

Откройте принтер в веб-интерфейсе CUPS, затем перейдите на страницу Administration ▸ Modify Printer (Администрирование ▸ Изменить принтер). Процесс настройки похож на установку нового принтера, за исключением того, что здесь отображаются текущие настройки принтера. На рис. 15.13 показано, как включить общий доступ к принтеру. (Обратите внимание: перед этим необходимо включить поддержку совместного доступа на странице Administration (Администрирование) в разделе Server (Сервер).)

Рис. 15.13. Изменение настроек установленного принтера

Комментарий

Вы можете изменить любые настройки, кроме имени принтера.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.10. Печать документов в файлы PDF

Задача

Сохранить веб-страницу или любой другой документ в формате PDF.

Решение

Откройте диалог File ▸ Print (Файл ▸ Печать) в любом приложении, и вы увидите вариант печати в файл PDF (рис. 15.14).

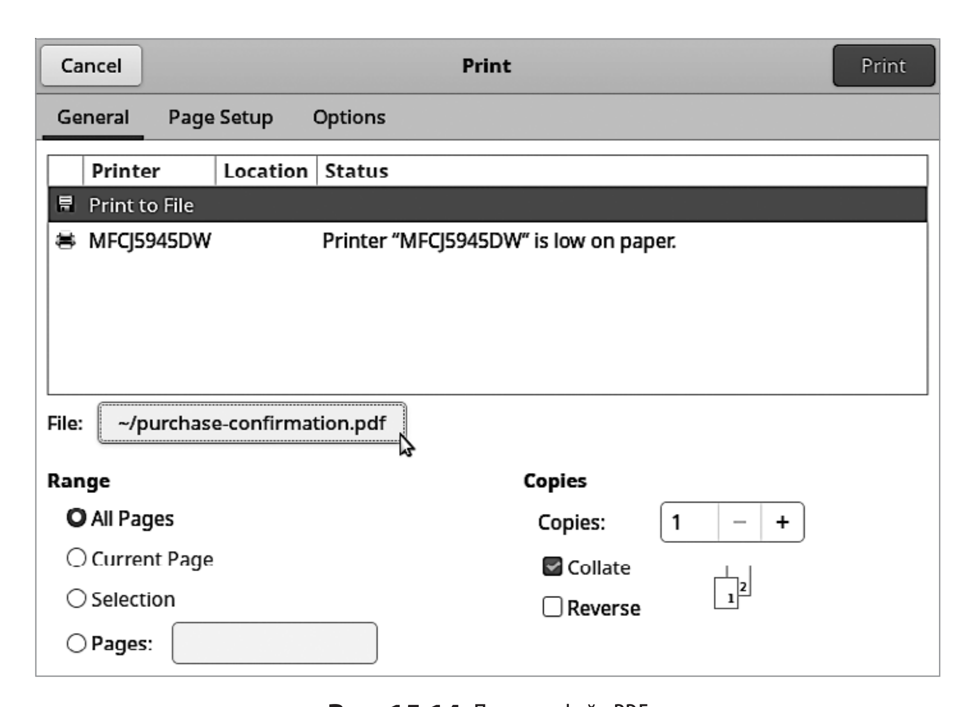


Рис. 15.14. Печать в файл PDF

При выборе этого варианта вам будут доступны все обычные параметры, такие как имя файла и его местоположение, поля, качество печати, цветная или монохромная и ориентация страницы. Диалоги принтера выглядят по-разному в разных приложениях; например, диалог печати в браузере Firefox включает предварительный просмотр документа. В других приложениях предварительный просмотр обычно открывается в отдельном окне после щелчка на специальной кнопке.

Комментарий

Печать в файл отлично подходит для сохранения веб-форм и квитанций, а также для создания PDF-файлов из документов любого типа.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

15.11. Устранение неполадок

Задача

Печать не работает! Как исправить проблему?

Решение

Ниже представлены наиболее типичные проблемы с принтерами в Linux.

- Если проблема с сетевым принтером, то убедитесь, что ваша сеть настроена правильно и ваш брандмауэр пропускает трафик к TCP-порту 631. Если у вас несколько сетей, то убедитесь, что принтер находится в той же сети, что и ваш компьютер.
- Если принтер подключен через USB, попробуйте подключить его к другому порту USB или другим кабелем.
- Убедитесь, что используете правильные драйверы, или попробуйте вариант печати без драйверов.
- Если демон CUPS управляется systemd, то попробуйте перезапустить демон:

  ```
  $ sudo systemctl restart cups.service
  ```

  Или перезагрузите систему. Попробуйте также выключить и включить принтер.

- Загляните в файлы журнала на странице веб-интерфейса CUPS; просмотрите оба журнала — журнал ошибок (error log) и журнал доступа (access log). Увеличьте уровень подробности журналирования до Debug (Отладка), чтобы получить максимум информации. (Нажмите кнопку Edit Configuration File (Редактировать конфигурационный файл) и настройте параметр `LogLevel debug`.)

Комментарий

Самый важный фактор — это использование принтеров с качественной поддержкой Linux. Наличие такой поддержки предотвратит большинство проблем.

Дополнительная информация

- Документация для CUPS (https://oreil.ly/OlCzV).

ГЛАВА 16

Управление локальной службой имен с помощью Dnsmasq и файла hosts

Dnsmasq (https://oreil.ly/MUa4U) — отличный сервер для служб имен в локальных сетях, который можно использовать как для системы доменных имен (Domain Name System, DNS), так и для динамического обнаружения хостов (Dynamic Host Discovery Protocol, DHCP). Dnsmasq также поддерживает протоколы BOOTP, PXE и TFTP, предназначенные для загрузки по сети и установки операционных систем с сетевого сервера. Dnsmasq поддерживает IPv4 и IPv6, обеспечивает локальное кэширование имен и действует как тупиковый резолвер (stub resolver).

В этой главе описывается настройка локальных служб DNS и DHCP с помощью Dnsmasq и файла `/etc/hosts`. Использование файла `/etc/hosts` — очень старый способ настройки DNS, основанный на описании соответствий имен хостов и IP-адресов в статическом файле. Файла `/etc/hosts` вполне достаточно для очень маленьких сетей.

Dnsmasq разрабатывался для организации служб имен в локальных сетях. Это легковесный и простой в настройке сервер, особенно по сравнению с BIND, доминирующим DNS-сервером, довольно тяжеловесным и требующим определенного обучения обращению с ним.

Dnsmasq и `/etc/hosts` отлично работают вместе. Dnsmasq читает записи из `/etc/hosts` в DNS.

Сервер DHCP в Dnsmasq автоматически интегрируется с DNS. Чтобы Dnsmasq создавал DNS-записи для ваших DHCP-клиентов, вы должны лишь настроить DHCP-клиентов, чтобы они отправляли свои имена хостов на DHCP-сервер, что и так сделано по умолчанию в большинстве дистрибутивов Linux.

Существует четыре типа DNS-серверов: рекурсивные резолверы, корневые серверы имен, серверы имен домена верхнего уровня (Top-Level Domain, TLD) и полномочные серверы имен.

Рекурсивные резолверы отвечают на запросы DNS. Тупиковый резолвер, такой как Dnsmasq и systemd-resolved, пересылает любые запросы, ответ на которые он не может получить из своего кэша, вышестоящему резолверу. Когда вы пытаетесь открыть страницу сайта, рекурсивный резолвер отыскивает DNS-информацию о сайте, посылая запросы DNS-серверам трех других типов. Рекурсивные резолверы кэшируют эту информацию, чтобы ускорить доступ к ней в будущем. Серверы имен вашего интернет-провайдера и такие службы, как OpenDNS (https://oreil.ly/oCRsV), Cloudflare (https://oreil.ly/9Fgqc) и Google Public DNS (https://oreil.ly/lc9ep), являются рекурсивными резолверами.

Существует 13 типов корневых серверов имен, разбросанных по всей планете. Всего в настоящее время имеется несколько сотен корневых серверов имен. Корневой сервер принимает запрос от рекурсивного резолвера, пересылает его соответствующему серверу имен домена верхнего уровня (TLD): .com, .net, .org, .me, .biz, .int, .biz, .gov, .edu и т. д. Все эти серверы и домены курирует корпорация по управлению доменными именами и IP-адресами (Internet Corporation for Assigned Names and Numbers, ICANN; https://icann.org).

Полномочные (authoritative) серверы имен служат источниками записей для домена и контролируются владельцем домена. Dnsmasq тоже может служить вашим полномочным сервером имен, хотя я рекомендую использовать BIND. Дополнительную информацию ищите в разделе Authoritative Configuration в руководстве `man 8 dnsmasq`.



Слишком много утилит службы имен

Дистрибутивы Linux все еще продолжают переходить на NetworkManager и systemd-resolved с устаревшего resolvconf, который долгое время был DNS-резолвером по умолчанию в системах Linux. Это несколько запутывает пользователей Linux из-за постоянных изменений и различий в разных дистрибутивах, выполняющих переход с разной скоростью. Обратите особое внимание на документацию, форумы и примечания к выпуску для вашего конкретного дистрибутива Linux.

У вас должна иметься возможность использовать Dnsmasq в качестве сервера DNS для NetworkManager, поскольку NetworkManager имеет для этого специализированный плагин. Но в некоторых дистрибутивах Linux данная связка пока работает некорректно (см. рецепт 16.5).

Вам не нужно запускать systemd-resolved на своем сервере Dnsmasq, поскольку он будет конфликтовать с Dnsmasq, состязаясь за контроль над системным тупиковым DNS-резолвером.

К тому времени, когда вы будете читать эти строки, ситуация может измениться к лучшему, но пока рецепты нацелены на то, чтобы быть надежными, а не ультрасовременными.

16.1. Простое разрешение имен с помощью файла /etc/hosts

Задача

Нужен простой и быстрый способ настроить разрешение имен без запуска сервера DNS.

Решение

Именно для этой цели предусмотрен файл `/etc/hosts`. Компьютеры в локальной сети должны иметь статические IP-адреса. Ниже показан пример содержимого данного файла для трех компьютеров:

```
127.0.0.1 localhost
::1 localhost ip6-localhost ip6-loopback
192.168.43.81 host1
192.168.43.82 host2
192.168.43.83 host3
```

Скопируйте эти записи на все три хоста, а затем попробуйте послать эхо-запрос друг другу по именам хостов, как в следующем примере:

```
host3:~$ ping -c2 host2
PING host2 (192.168.43.82) 56(84) bytes of data.
64 bytes from host2 (192.168.43.82): icmp_seq=1 ttl=64 time=3.00 ms
64 bytes from host2 (192.168.43.82): icmp_seq=2 ttl=64 time=3.81 ms

--- host2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 3.001/3.403/3.806/0.402 ms
```

Файл `/etc/hosts` также может управлять доменными именами, поэтому вы можете дать своей локальной сети какое-нибудь классное доменное имя. В следующих примерах это `sqr3l.nut`. Сначала введите IP-адрес, затем полное доменное имя (Fully Qualified Domain Name, FQDN) и потом имя хоста:

```
127.0.0.1 localhost
::1 localhost ip6-localhost ip6-loopback
192.168.43.81 host1.sqr3l.nut host1
192.168.43.82 host2.sqr3l.nut host2
192.168.43.83 host3.sqr3l.nut host3
```

Теперь хосты смогут подключаться друг к другу, используя простые имена хостов или полные доменные имена, такие как `host1` или `host1.sqr3l.nut`.



Общие и индивидуальные записи в файле /etc/hosts

Вы можете иметь как общие, так и личные записи в /etc/hosts. Все, чем вы хотите поделиться, следует скопировать на все соответствующие хосты. Все остальное в вашем файле hosts, что не копируется на другие хосты, будет доступно только вам. Дополнительно об этом рассказывается в рецепте 16.2.

Комментарий

Записи `127.0.0.1 localhost` и `::1 localhost ip6-localhost ip6-loopback` являются обязательными. У вас они могут выглядеть немного иначе, но, как бы ни выглядели, не удаляйте их. Они назначаются петлевому устройству — специальному виртуальному сетевому интерфейсу, который система Linux использует для связи с самой собой.

Вы можете посылать по этим именам эхо-запросы и использовать для подключения к локальным серверам. Например, открывая веб-страницу администрирования CUPS, вы задействуете петлевое устройство. Введите `127.0.0.1:631` или `localhost: 631` в адресной строке браузера, чтобы открыть ее (рис. 16.1).

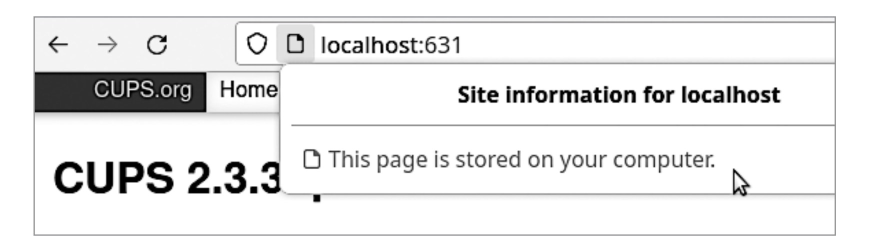


Рис. 16.1. Доступ к локальной веб-странице через петлевой сетевой интерфейс

Виртуальный сетевой интерфейс для петлевого устройства называется `lo`. Воспользуйтесь командой `ip`, чтобы получить информацию о нем:

```
$ ip addr show dev lo
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
  default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
```

Для работы петлевого устройства системе не требуется наличие физического сетевого интерфейса.

Выполните команду `hostname`, чтобы убедиться, что ваша конфигурация верна. Проверьте имя хоста своего компьютера:

```
$ hostname
host1
```

Проверьте FQDN:

```
$ hostname -f
host1.sqr3l.nut
```

Проверьте имя домена:

```
$ hostname -d
sqr3l.nut
```

Подход с использованием файла `/etc/hosts` не масштабируется, но в маленьких сетях его может быть более чем достаточно для организации локальной службы имен DNS.

Дополнительная информация

- `man 5 hosts`
- `man 8 ping`
- Рецепт 16.2.

16.2. Использование файла /etc/hosts для тестирования и блокировки надоедливых сайтов

Задача

Вы работаете с серверами разработки и хотите организовать разрешение их имен. Или просто хотите заблокировать надоедливые сайты.

Решение

Предположим, что сервер разработки, с которым вы работаете, имеет имя `dev.stashcat.com`. Добавьте для него запись в свой файл `/etc/hosts`:

```
192.168.10.15 dev.stashcat.com
```

Вам не нужно беспокоить сетевого администратора или настраивать свой DNS-сервер, намного проще создавать и удалять записи в `/etc/hosts` по мере необходимости.

Еще один забавный трюк — отображение имен надоедливых сайтов в фиктивные IP-адреса:

```
12.34.56.78 badsite.com
12.34.56.78 www.badsite.com
```

Это сделает сайт недоступным с вашего компьютера. В большинстве инструкций рекомендуется использовать петлевой адрес 127.0.0.1, и это вполне работоспособный прием, но я предпочитаю держать раздражающие сайты подальше. Кстати, один и тот же фиктивный IP-адрес можно применять для нескольких надоедливых сайтов.

Если ваш браузер все так же отыскивает сайт после добавления его в файл `/etc/hosts`, то попробуйте очистить кэш браузера и повторить попытку.

Комментарий

Запуская сервер Dnsmasq, имейте в виду, что все записи в файле `/etc/hosts` на сервере Dnsmasq будут применяться ко всем клиентам Dnsmasq, поэтому не помещайте свой сервер имен на компьютер разработки.

В Linux есть несколько источников информации DNS, и `/etc/hosts` читается первым. Порядок определяется параметром `hosts` в файле `/etc/nsswitch.conf`. Следующий пример взят из Ubuntu 20.04:

```
hosts: files mdns4_minimal [NOTFOUND=return] dns mymachines
```

`files` — это файл `/etc/hosts`.

`mdns4_minimal` использует службу Avahi автоматического обнаружения сетевых сервисов.

`[NOTFOUND=return]` означает, что если `mdns4_minimal` работает, но запрошенный хост не был найден, то поиск имени следует прервать и вернуть признак ошибки. Если служба `mdns4_minimal` не найдена, то поиск продолжается.

`dns` означает любой доступный сервер DNS.

`mymachines` означает службу systemd-machined, которая следит за имеющимися локальными виртуальными машинами и контейнерами.

Используя сервер Dnsmasq, поместите `files dns` в начало списка.

Дополнительная информация

- `man 5 hosts`
- `man 5 nsswitch.conf`
- `man 8 systemd-machined.service`

16.3. Поиск всех серверов DNS и DHCP в своей сети

Задача

Узнать, имеются ли в локальной сети серверы DNS и DHCP, отличные от Dnsmasq.

Решение

Проверьте свою локальную сеть с помощью `nmap`. В следующем примере выполняется поиск всех открытых TCP-портов в локальной сети и обнаруживается открытый TCP-порт 53, который используется службой DNS. На это указывает текст `53/tcp open domain`:

```
$ sudo nmap --open 192.168.1.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2021-05-23 13:25 PDT
[...]
Nmap scan report for dns-server.sqr3l.nut (192.168.1.10)
Host is up (0.12s latency).
Not shown: 998 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit

PORT   STATE SERVICE
22/tcp open  ssh
53/tcp open  domain
[...]

Nmap done: 256 IP addresses (3 hosts up) scanned in 81.38 seconds
```

По умолчанию `nmap` проверяет только TCP-порты. Но DNS-серверы прослушивают TCP- и UDP-порт 53, а DHCP — UDP-порт 67. Следующий пример ищет только открытые порты UDP 53 и 67:

```
$ sudo nmap -sU -p 53,67 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-27 18:05 PDT

Nmap scan report for dns-server.sqr3l.nut (192.168.1.10)
Host is up (0.085s latency).

PORT   STATE         SERVICE
53/udp open          domain
67/udp open|filtered dhcps

Nmap done: 256 IP addresses (3 hosts up) scanned in 13.85 seconds
```

Сканер `nmap` нашел только один сервер DNS/DHCP — на `dns-server.sqr3l.nut`.

Следующая команда ищет все открытые порты TCP и UDP в сети:

```
$ sudo nmap -sU -sT 192.168.1.0/24
```

Поиск займет несколько минут, после чего у вас будет список активных служб на всех узлах сети, включая все службы, работающие на нестандартных портах.

Комментарий

Будьте очень осторожны со сканированием портов и прибегайте к нему только в сетях, на сканирование которых у вас есть разрешение. Во многих сетях сканирование портов рассматривается как враждебный акт, например как попытка использовать уязвимости.

Наличие нескольких серверов имен может приводить к конфликтам, и в любом случае важно знать, работают ли ваши пользователи на каких-либо серверах.

В большинстве дистрибутивов Linux пакет *nmap* нужно устанавливать специально.

Дополнительная информация

- `man 1 nmap`

16.4. Установка Dnsmasq

Задача

Установить Dnsmasq и все необходимое.

Решение

Установите пакет *dnsmasq*. В этом рецепте сервер Dnsmasq называется `dns-server`. Для настройки сервера DNS мы будем использовать как Dnsmasq, так и файл `/etc/hosts`.

После установки остановите службу Dnsmasq, если она была запущена:

```
$ systemctl status dnsmasq.service
● dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
    Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor
```

```
     preset: enabled)
     Active: active (running) since Mon 2021-05-24 05:49:36 PDT; 6h ago
[...]
$ sudo systemctl stop dnsmasq.service
```

Назначьте своему серверу Dnsmasq статический IP-адрес, если он его еще не имеет. Это можно сделать на графической панели управления NetworkManager (`nm-connection-editor`) или в командной строке, запустив команду `nmcli`.

Следующий пример демонстрирует, как с помощью `nmcli` отыскать свои активные подключения к сети:

```
$ nmcli connection show --active
NAME       UUID                        TYPE      DEVICE
1local     3e348c97-4c5f-4bbf-967e  wifi      wlan1
1wired     0460d735-e14d-3c3f-92c0  ethernet  eth1
```

Затем назначьте статический IP-адрес своему серверу DNS, указав в команде имя подключения (из столбца `NAME`):

```
$ nmcli con mod "1wired" \
  ipv4.addresses "192.168.1.30/24" \
  ipv4.gateway "192.168.1.1" \
  ipv4.method "manual"
```

Теперь перезапустите NetworkManager:

```
$ sudo systemctl restart NetworkManager.service
```

Далее проверьте — запущена ли служба systemd-resolved.service:

```
$ systemctl status systemd-resolved.service
```

Если запущена, то прочитайте рецепт 16.5, прежде чем приступать к настройке Dnsmasq, а также изучите настройки NetworkManager на своем сервере Dnsmasq.

Комментарий

В разных дистрибутивах Linux система инициализации systemd реализована по-разному. Например, openSUSE Leap 15.2 не использует службу systemd-resolved.service, поэтому в данном дистрибутиве не нужно вносить какие-либо изменения в настройки systemd, чтобы разрешить Dnsmasq управлять разрешением имен в вашей локальной сети. В Fedora 33 и выше, а также в Ubuntu 17.04 и выше служба systemd-resolved.service запускается автоматически и ее нужно отключить на сервере Dnsmasq.

Дополнительная информация

- Рецепт 16.5.
- Dnsmasq (https://oreil.ly/vvfHg).

16.5. Устранение конфликтов между systemd-resolved с NetworkManager и Dnsmasq

Задача

Служба systemd-resolved и NetworkManager конфликтуют с Dnsmasq, и этот конфликт нужно устранить.

Решение

Проверьте, запущена ли служба systemd-resolved.service:

```
$ systemctl status systemd-resolved.service

● systemd-resolved.service - Network Name Resolution
     Loaded: loaded (/usr/lib/systemd/system/systemd-resolved.service; enabled;
     vendor preset: enabled)
     Active: active (running) since Sat 2021-05-22 12:57:34 PDT; 1min 21s ago
[...]
```

В данном случае служба запущена. Она отлично подходит на роль тупикового резолвера DNS для клиентских машин, но не для DNS-серверов. Отключите ее:

```
$ sudo systemctl stop systemd-resolved.service
$ sudo systemctl disable systemd-resolved.service
```

Затем отыщите файл `/etc/resolv.conf`, который должен быть символической ссылкой:

```
$ ls -l /etc/resolv.conf
lrwxrwxrwx 1 root root 39 May 21 20:38 /etc/resolv.conf ->
   ../run/systemd/resolve/stub-resolv.conf
```

Эта символическая ссылка управляется службой systemd-resolved.service. Чтобы отключить управление из данной службы, удалите символическую ссылку и создайте текстовый файл с тем же именем:

```
$ sudo rm /etc/resolv.conf
$ sudo touch /etc/resolv.conf
```

Теперь, когда /etc/resolv.conf — это обычный файл, а не символическая ссылка, им будет управлять NetworkManager. Откройте конфигурационный файл NetworkManager и найдите раздел [main], затем добавьте или измените значение параметра dns= на none:

```
$ sudo nano /etc/NetworkManager/NetworkManager.conf

[main]
dns=none
```

Добавьте в файл /etc/resolv.conf адреса локального хоста IPv4 и IPv6 сервера Dnsmasq и свой локальный домен, если он у вас есть:

```
search sqr3l.nut
nameserver 127.0.0.1
nameserver ::1
```

Затем перезагрузите и настройте Dnsmasq.

Комментарий

NetworkManager и служба systemd-resolved хорошо подходят для клиентских машин. Но на сервере DNS контроль над файлом /etc/resolv.conf должен иметь только Dnsmasq и быть единственным тупиковым резолвером.

Дополнительная информация

- man 8 systemd-resolved.service
- man 8 networkmanager

16.6. Настройка Dnsmasq на роль сервера DNS для локальной сети

Задача

Настроить Dnmasq на роль сервера DNS для локальной сети.

Решение

Любые хосты, перечисленные в файле /etc/hosts, должны иметь статические IP-адреса, и Dnsmasq автоматически добавит их в таблицу DNS. Для начала

добавьте свой сервер Dnsmasq. В следующем примере содержимое `/etc/hosts` включает сервер Dnsmasq, сервер резервного копирования и внутренний веб-сервер:

```
127.0.0.1 localhost
::1 localhost ip6-localhost ip6-loopback
192.168.43.81 dns-server
192.168.43.82 backups
192.168.43.83 https
```



Настройка статических адресов хостов в DHCP

В рецепте 16.12 вы узнаете, как организовать назначение статических IP-адресов в DHCP вместо файла /etc/hosts.

Теперь можно приступать к настройке Dnsmasq. Переименуйте конфигурационный файл с настройками по умолчанию, чтобы начать с нового пустого файла, а прежний использовать для справки:

```
$ sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf-old
$ sudo nano /etc/dnsmasq.conf
```

Скопируйте следующие настройки конфигурации, подставив во второй параметр `listen-address` IP-адрес своего сервера, и укажите свое доменное имя. Вышестоящие серверы имен в этом примере — серверы OpenDNS, но вы можете использовать любые другие вышестоящие серверы имен, какие пожелаете. По умолчанию Dnsmasq ищет имена в файле `/etc/resolv.conf`, но вообще лучше зафиксировать такой порядок поиска явно:

```
# глобальные параметры
resolv-file=/etc/resolv.conf
domain-needed
bogus-priv
expand-hosts
domain=sqr3l.nut
local=/sqr3l.nut/
listen-address=127.0.0.1
listen-address=192.168.43.81

# вышестоящие серверы имен
server=208.67.222.222
server=208.67.220.220
```

Запустите синтаксическую проверку файла:

```
$ dnsmasq --test
dnsmasq: syntax check OK.
```

Эта проверка не обнаруживает ошибки в конфигурации — только опечатки. Запустите Dnsmasq, и если в настройках есть какие-то ошибки, то он не запустится. В следующем примере показан успешный запуск:

```
$ sudo systemctl start dnsmasq.service
$ systemctl status dnsmasq.service
● dnsmasq.service - dnsmasq - A lightweight DHCP and caching DNS server
     Loaded: loaded (/lib/systemd/system/dnsmasq.service; enabled; vendor preset:
     enabled)
     Active: active (running) since Mon 2021-05-24 17:13:48 PDT; 1min 0s ago
    Process: 11023 ExecStartPre=/usr/sbin/dnsmasq --test (code=exited,
     status=0/SUCCESS)
    Process: 11024 ExecStart=/etc/init.d/dnsmasq systemd-exec (code=exited,
     status=0/SUCCESS)
    Process: 11033 ExecStartPost=/etc/init.d/dnsmasq systemd-start-resolvconf
     (code=exited, status=0/SUCCESS)
   Main PID: 11032 (dnsmasq)
      Tasks: 1 (limit: 18759)
     Memory: 2.5M
     CGroup: /system.slice/dnsmasq.service
             └─11032 /usr/sbin/dnsmasq -x /run/dnsmasq/dnsmasq.pid -u dnsmasq -7
              /etc/dnsmasq.d,.dpkg-dist,.dpkg-old,.dpkg-new --local->

May 24 17:13:48 dns-server systemd[1]: Starting dnsmasq - A lightweight DHCP and
 caching DNS server...
May 24 17:13:48 dns-server dnsmasq[11023]: dnsmasq: syntax check OK.
May 24 17:13:48 dns-server systemd[1]: Started dnsmasq - A lightweight DHCP and
 caching DNS server.
```

Проверьте работу сервера Dnsmasq, попробовав определить IP-адрес хоста сервера по его имени и FQDN с помощью команды `nslookup`:

```
$ nslookup dns-server
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   dns-server
Address: 192.168.43.81

$ nslookup dns-server.sqr3l.nut
Server:         127.0.0.1
Address:        127.0.0.1#53
```

```
Name:   dns-server.sqr3l.nut
Address: 192.168.43.81

$ nslookup 192.168.43.81
18.43.168.192.in-addr.arpa      name = host1.sqr3l.nut.
```

Используйте команду `ss` для проверки открытых портов. В следующем примере столбцы `Recv+Q`, `Send+Q` и `Peer Address:Port` были опущены для ясности:

```
$ sudo ss -lp "sport = :domain"
Netid  State   Local Address:Port    Process
udp    UNCONN     127.0.0.1:domain   users:(("dnsmasq",pid=1531,fd=8))
udp    UNCONN  192.168.1.10:domain   users:(("dnsmasq",pid=1531,fd=6))
tcp    LISTEN     127.0.0.1:domain   users:(("dnsmasq",pid=1531,fd=9))
tcp    LISTEN  192.168.1.10:domain   users:(("dnsmasq",pid=1531,fd=7))
```

Вы должны увидеть адрес своего сервера, адрес локального хоста и только `dnsmasq` в столбце `Process`. Добавьте параметр `-r`, чтобы увидеть имена хостов вместо IP-адресов.

Если все эти команды выполнились успешно, значит, ваша конфигурация верна.

Комментарий

Если Dnsmasq не запускается, то запустите `journalctl -ru dnsmasq`, чтобы узнать причину. (Если журналы Dnsmasq отправляются куда-то еще, посмотрите там; см. рецепт 16.14.)

Команда `nslookup` входит в состав пакета *bindutils*.

`ss` (socket statistics — «статистика сокетов») входит в состав пакета *iproute2*.

Если команды `nslookup` потерпели фиаско, то попробуйте перезапустить сеть, а затем перезапустить Dnsmasq. Если вас вновь постигла неудача, то перезагрузитесь. Если и это не помогло, то тщательно проверьте все настройки.

`domain-needed` не позволяет Dnsmasq пересылать запросы на поиск простых имен хостов вышестоящим серверам имен. Если имя не указано в файле `/etc/hosts` или в DHCP, то возвращается ответ «не найдено». Это предотвращает утечку запросов адресов из локальной сети во внешний мир и, возможно, получение неправильного ответа, если вдруг имя домена вашей локальной сети совпадет с именем общедоступного домена.

`bogus-priv` блокирует обратный поиск поддельных частных адресов. В ответ на все попытки обратного поиска в диапазонах частных IP-адресов, отсутствующих

в файле `/etc/hosts` или в файле аренды DHCP, возвращается ответ «нет такого домена» вместо передачи вышестоящим серверам имен.

`expand-hosts` автоматически добавляет имя частного домена к простым именам хостов в файле `/etc/hosts`.

`domain =` — имя локального домена.

`local = / [домен] /` требует от Dnsmasq разрешать запросы для локального домена напрямую, а не пересылать их вышестоящим серверам.

Дополнительная информация

- `man 5 hosts`
- Dnsmasq (https://oreil.ly/vvfHg).

16.7. Настройка поддержки DNS и DHCP в firewalld

Задача

Открыть в брандмауэре доступ к серверу Dnsmasq, чтобы клиенты в локальной сети могли пользоваться им.

Решение

Откройте порты TCP и UDP с номером 53 (DNS) и порт UDP с номером 67 (DHCP). Если в качестве брадмауэра используется firewalld, то выполните следующую команду:

```
$ sudo firewall-cmd --permanent --add-service=\{dns,dhcp\}
```

Комментарий

При возникновении проблем с доступом в первую очередь необходимо проверить настройки брандмауэра.

Дополнительная информация

- Глава 14.

16.8. Тестирование сервера Dnsmasq с машины клиента

Задача

Протестировать работу нового DNS-сервера Dnsmasq с клиентского компьютера.

Решение

Воспользуйтесь командой `dig` на любом хосте в сети и отправьте запрос по IP-адресу вашего сервера Dnsmasq, чтобы получить информацию о любом сайте:

```
$ dig @192.168.1.10 oreilly.com
; <<>> DiG 9.16.6 <<>> @192.168.1.10 oreilly.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29387
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;oreilly.com.                   IN      A

;; ANSWER SECTION:
oreilly.com.            240     IN      A       199.27.145.65
oreilly.com.            240     IN      A       199.27.145.64

;; Query time: 108 msec
;; SERVER: 192.168.1.10#53(192.168.1.10)
;; WHEN: Mon May 24 17:49:32 PDT 2021
;; MSG SIZE  rcvd: 72
```

Успешность теста подтверждает строка `status: NOERROR` и IP-адрес сервера Dnsmasq в строке `SERVER`.

Комментарий

Работу сервера имен также можно протестировать, использовав имя хоста и полное доменное имя вашего сервера вместо IP-адреса.

```
$ dig @dns-server oreilly.com
$ dig @dns-server.sqr3l.nut oreilly.com
```

Дополнительная информация

- man 1 dig

16.9. Управление службой DHCP с помощью Dnsmasq

Задача

Сервер имен настроен и работает, и теперь нужно настроить DHCP.

Решение

Это простая задача. Добавьте следующие строки в файл `/etc/dnsmasq.conf`, чтобы определить единый пул адресов, подставив свои адреса:

```
# Диапазон DHCP
dhcp-range=192.168.1.25,192.168.1.75,12h
dhcp-lease-max=25
```

Перезапустите Dnsmasq:

```
$ sudo systemctl restart dnsmasq.service
```

Попробуйте получить адрес на компьютере в локальной сети, но сначала убедитесь, что он настроен на получение IP-адреса через DHCP:

```
$ nmcli con show --active
NAME    UUID                        TYPE     DEVICE
1net    de7c00e7-8e4d-45e6-acaf  ethernet eth0

$ nmcli con show 1net | grep ipv..method
ipv4.method:             auto
ipv6.method:             auto
```

Вывод значения `auto` подтверждает, что данный компьютер использует DHCP. (Если вместо `auto` вы увидите `manual`, то это значит, что компьютер использует статические настройки.) Остановите сетевой интерфейс и снова запустите его:

```
$ sudo nmcli con down 1net
Connection '1net' successfully deactivated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/11
```

```
$ sudo nmcli con up 1net
Connection successfully activated (D-Bus active path: /org/freedesktop/NetworkMan
ager/ActiveConnection/15)
```

Проверьте содержимое журнала сервера Dnsmasq:

```
$ journalctl -ru dnsmasq
-- Logs begin at Sun 2021-02-28 14:35:01 PST, end at Mon 2021-05-31 17:36:04
 PDT. --
May 31 17:34:56 dns-server dnsmasq-dhcp[8080]: DHCPACK(eth0) 192.168.1.45
 9c:ef:d5:fe:01:7c client2
May 31 17:34:56 dns-server dnsmasq-dhcp[8080]: DHCPREQUEST(eth0) 192.168.1.45
9c:ef:d5:fe:01:7c
```

В данном случае записи в журнале сообщают, что назначение IP-адреса сервером `dns-server` клиенту `client2` прошло успешно.

Комментарий

Вместо `nmcli` можно использовать апплет панели NetworkManager или запустить команду `nm-connection-editor`, чтобы открыть графический диалог настройки NetworkManager, в котором можно отключиться и подключиться одним щелчком кнопкой мыши (рис. 16.2).

Большинство дистрибутивов Linux задействуют NetworkManager для управления клиентом DHCP. Если у вас это не так, то, вероятнее всего, используется команда `dhclient`. Найдите конфигурационный файл `dhclient.conf`, если он существует, а затем запросите новую аренду IP-адреса с помощью команды `dhclient`:

```
$ sudo dhclient -v
Internet Systems Consortium DHCP Client 4.3.6-P1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/9c:ef:d5:fe:01:7c
Sending on   LPF/eth0/9c:ef:d5:fe:01:7c
Sending on   Socket/fallback
DHCPREQUEST on eth0 to 255.255.255.255 port 67 (xid=0xec8923)
DHCPACK from 192.168.1.10 (xid=0xec8923)
bound to 192.168.1.27 -- renewal in 1415 seconds.
```

Через DHCP можно отправлять клиентам дополнительную информацию, необходимую для доступа к сетевым службам. Подробнее об этом рассказывается в рецепте 16.10.

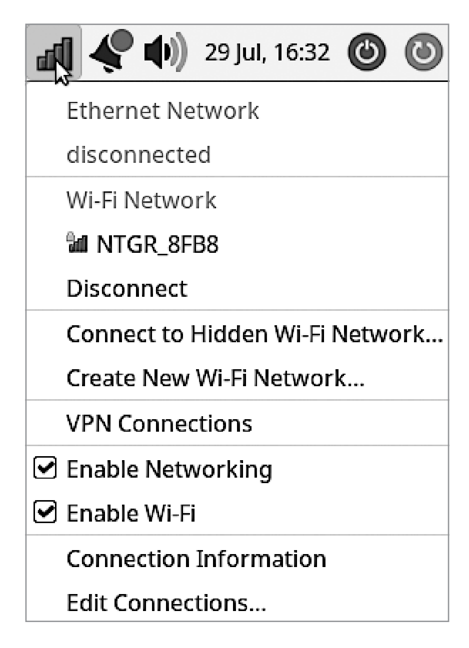


Рис. 16.2. Управление подключением к сети с помощью команды nm-connection-editor

`dhcp-range=192.168.1.25,192.168.10.75,24h` определяет диапазон из 50 адресов, доступных для аренды на срок до 24 часов. Этот диапазон не должен включать IP-адреса вашего сервера Dnsmasq и любых других хостов со статическими IP-адресами. Продолжительность аренды можно задавать в секундах, минутах или часах. По умолчанию адрес арендуется на один час, минимальная продолжительность аренды — две минуты. Если вам нужна бессрочная аренда, то не указывайте ее продолжительность.

`dhcp-lease-max=25` определяет, сколько адресов может быть арендовано одновременно. Вы можете иметь большой пул адресов и ограничивать количество одновременно действующих аренд.

Дополнительная информация

- Рецепт 16.10.
- Dnsmasq (https://oreil.ly/vvfHg).
- `man 8 dhclient`

16.10. Передача важной информации о службах через DHCP

Задача

Организовать передачу клиентам настроек доступа к различным серверам в локальной сети через DHCP.

Решение

Некоторые настройки служб, такие как маршрут по умолчанию к интернет-шлюзу, DNS-серверу и NTP-серверу, могут автоматически передаваться клиентам в локальной сети. В следующих примерах показано, что добавить в файл `/etc/dnsmasq.conf` для автоматической передачи настроек некоторых служб.

Настройка маршрутизатора по умолчанию:

```
dhcp-option=3,192.168.1.1
```

Настройка сервера DNS:

```
dhcp-option=6,192.168.1.10
```

Ниже представлен пример, показывающий, как передать клиентам путь к локальному серверу NTP:

```
dhcp-option=42,192.168.1.11
```

Как узнать, какие номера параметров использовать? Следующая команда выведет их все:

```
$ dnsmasq --help dhcp
Known DHCP options:
  1 netmask
  2 time-offset
  3 router
  6 dns-server
  7 log-server
  9 lpr-server
  [...]
```

Комментарий

Команда `dnsmasq --help dhcp` покажет известные номера параметров конфигурации DHCPv4. Дополнительно о параметрах конфигурации DHCPv4 рассказывается в подразделе «Комментарий» рецепта 16.11.

Дополнительная информация

- Dnsmasq (https://oreil.ly/vvfHg).

16.11. Создание зон DHCP для подсетей

Задача

Имеется две подсети, и требуется настроить Dnsmasq для распространения в них отличающихся настроек, таких как адреса маршрутизаторов и серверов по умолчанию.

Решение

Определите зоны с любыми именами по своему выбору, например `zone1` и `zone2`, и задайте диапазоны их адресов:

```
dhcp-range=zone1,192.168.50.20,192.168.50.120
dhcp-range=zone2,192.168.60.20,192.168.60.50,24h
```

Зоны имеют разные маршрутизаторы:

```
dhcp-option=zone1,3,192.168.50.1
dhcp-option=zone2,3,192.168.60.2
```

Используют один сервер DNS:

```
dhcp-option=zone1,6,192.168.1.10
dhcp-option=zone2,6,192.168.1.10
```

Зона `zone2` имеет сервер NTP:

```
dhcp-option=zone2,42,192.168.60.15
```

Комментарий

Лишь часть параметров DHCP имеет практическую пользу. Есть параметры очень старые, а некоторые кажутся очень загадочными, например:

> `параметр default-url строка;`
>
> Формат и назначение данного параметра не описаны ни в одном стандарте, но заявлено, что он используется Apple Computer. Неизвестно, что дает клиентам этот параметр. Используйте его на свой страх и риск.
>
> *man 5 dhcp-options*

Поддержка многих из них на стороне клиентов непоследовательна. Я использую только настройки NTP, маршрутизаторов и серверов DNS.

Дополнительная информация

- `man 5 dhcp`
- Dnsmasq (https://oreil.ly/vvfHg).

16.12. Назначение статических IP-адресов с помощью DHCP

Задача

Максимально централизовать назначение IP-адресов, в том числе и статических.

Решение

Используйте параметр `dhcp-host` в файле `/etc/dnsmasq.conf`. Определите имя хоста клиентского компьютера и назначьте требуемый IP-адрес из блока адресов вашей локальной сети. (Для статических адресов не обязательно применять адреса из диапазона адресов DHCP, определенного с помощью параметра `_dhcp-range=*` в `/etc/dnsmasq.conf`.) В следующем примере хосту `server2` назначается адрес, принадлежащий сети 192.168.3.0/24:

```
dhcp-host=server2,192.168.3.45
```

Перезапустите Dnsmasq, после этого, когда `server2` запросит адрес, он получит адрес, указанный в параметре `dhcp-host=`.

Добавьте несколько параметров `dhcp-host=`, чтобы настроить назначение статических адресов нескольким клиентам.

Вместо имени хоста можно также использовать MAC-адреса клиентов.

Комментарий

В общем случае централизация рутинной административной работы помогает экономить время и избавляет от проблем.

Дополнительная информация

- Dnsmasq (https://oreil.ly/vvfHg).

16.13. Настройка клиентов DHCP для автоматического создания записей в DNS

Задача

Необходимо, чтобы сведения о клиентах DHCP автоматически вводились в DNS с помощью Dnsmasq.

Решение

Единственное, что требуется от клиентов, — отправить свои имена хостов DHCP-серверу Dnsmasq, который используется по умолчанию в большинстве Linux.

Предположим, что клиент DHCP в локальном домене `sqr3l.nut` имеет имя хоста `client4`. Когда `client4` запускается, он получает свой IP-адрес и другую сетевую информацию от Dnsmasq. Dnsmasq, в свою очередь, получает имя хоста `client4` и вводит его в DNS. После этого другие хосты в сети смогут обращаться к клиенту по именам `client4` и `client4.sqr3l.nut`.

В файле `/etc/hosts` не должно быть повторяющихся записей для `client4`.

Есть три разных способа проверить конфигурацию клиента DHCP: в файле `dhclient.conf`, на графической панели настройки NetworkManager (*nm-connection-editor*) и с помощью команды `nmcli`.

Сначала проверьте настройки службы dhclient, которая в течение многих лет использовалась в Linux по умолчанию. В большинстве дистрибутивов Linux конфигурационный файл находится в `/etc/dhcp/dhclient.conf`. Найдите следующую строку, которая автоматически определяет имя хоста и отправляет его на сервер DHCP:

```
send host-name = gethostname();
```

Или строку, явно определяющую имя хоста:

```
send host-name = myhostname
```

Если в системе нет файла `dhclient.conf`, то вашим клиентом DHCP управляет NetworkManager. Убедиться в этом можно, запустив графическую панель nm-connection-editor (рис. 16.3).

Если выбран метод подключения Automatic (DHCP) (Автоматически), значит, NetworkManager отправляет имя хоста на сервер DHCP. Если выбран метод Automatic (addresses only) (Автоматически (только адреса)), значит, имя хоста не отправляется серверу DHCP, а только загружаются настройки DNS.

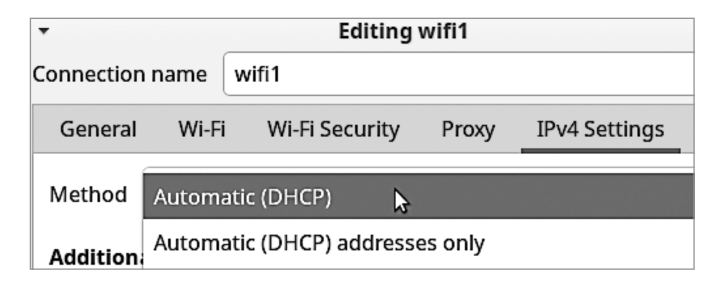


Рис. 16.3. NetworkManager посылает имя хоста клиента на сервер DHCP

Кроме того, вы можете использовать команду `nmcli`. Сначала найдите активное сетевое соединение:

```
$ nmcli connection show --active
NAME    UUID                                  TYPE     DEVICE
wifi1   3e348c97-4c5f-4bbf-967e-7624f3e1e4f0  wifi     wlan1
```

Затем проверьте, посылает ли клиент имя хоста серверу DHCP. Следующий пример подтверждает, что посылает:

```
$ nmcli connection show wifi1 | grep send-hostname
ipv4.dhcp-send-hostname:                yes
ipv6.dhcp-send-hostname:                yes
```

Если вы увидите `no` (нет), то выполните следующие команды, чтобы включить отправку. После этого перезагрузите конфигурацию:

```
$ sudo nmcli con mod wifi1 ipv4.dhcp-send-hostname yes
$ sudo nmcli con mod wifi1 ipv6.dhcp-send-hostname yes
$ sudo nmcli con reload
```

Комментарий

Если вы предпочитаете инструменты с графическим интерфейсом, то используйте графическую панель настройки NetworkManager, nm-connection-editor, а не другой графический инструмент, такой как сетевой модуль на панели управления GNOME, поскольку nm-connection-editor предлагает наиболее полный набор параметров настройки и одинаковый во всех дистрибутивах Linux.

Дополнительная информация

- `man 1 nmcli`
- `man 1 nmcli-examples`
- `man 5 nm-settings`

16.14. Управление журналированием в Managing

Задача

Dnsmasq может отправлять свои сообщения в файл по вашему выбору, используя устаревший демон syslog вместо journalctl, и вы хотите узнать, какой вариант лучше.

Решение

Выбор метода совершенно неважен: и в том и в другом случае сохраняется одна и та же информация. По умолчанию журналирование выполняется в журнал systemd.

Иногда удобно изолировать журналы Dnsmasq в отдельном каталоге, например `/var/log/dnsmasq/dnsmasq.log`. Используйте параметр `log-feature=` в файле `/etc/dnsmasq.conf`, чтобы указать файл журнала для использования, а затем перезапустите Dnsmasq. Файл должен уже существовать, иначе Dnsmasq не запустится.

Если не настроить ротацию журналов, то ваш файл журнала может стать очень большим. Следующий пример показывает, как настроить простую еженедельную ротацию в `/etc/logrotate.d/dnsmasq`:

```
/var/log/dnsmasq/dnsmasq.log {
    missingok
    compress
    notifempty
    rotate 4
    weekly
    create
    }
```

Проверьте настройки с помощью команды `logrotate`:

```
$ sudo /etc/logrotate.conf --debug
[...]
rotating pattern: /var/log/dnsmasq/dnssmasq.log  weekly (4 rotations)
empty log files are not rotated, old logs are removed
switching euid to 0 and egid to 4
considering log /var/log/dnsmasq/dnssmasq.log
Creating new state
  Now: 2021-06-01 13:08
  Last rotated at 2021-06-01 13:00
  log does not need rotating (log has been already rotated)
switching euid to 0 and egid to 0
[...]
```

Отсутствие сообщений об ошибках говорит, что все в порядке.

Комментарий

systemd поддерживает оба демона, journalctl и syslog. Вероятно, они будут существовать вместе еще очень долго, поэтому вы можете настроить журналирование любым удобным для себя способом.

Дополнительная информация

- `man 8 rsyslog`
- Dnsmasq (https://oreil.ly/vvfHg).
- `man 1 journalctl`
- Глава 20.

16.15. Настройка подстановочных доменов

Задача

Создать подстановочный домен в Dnsmasq, чтобы запросы к поддоменам в этом домене разрешались без ручного добавления поддоменов в DNS.

Решение

Используйте параметр `address` в файле `/etc/dnsmasq.conf`, чтобы создать домен верхнего уровня (Top-Level Domain, TLD):

```
address=/wildcard.net/192.168.1.35
```

Перезапустите Dnsmasq, затем выполните проверку с помощью команды `nslookup`:

```
$ sudo systemctl restart dnsmasq.service
$ nslookup foo.wildcard.net
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:    foo.wildcard.net
Address: 192.168.1.35
```

Имя `foo.wildcard.net` было разрешено, значит, все работает.

Комментарий

Используйте возможность подстановки доменов в DNS с осторожностью. Они могут пригодиться при разработке сложных служб, таких как Kubernetes. При этом всегда используйте диапазоны адресов, не пересекающиеся с диапазоном адресов на сервере имен вашей локальной сети и доступные только для клиентов локальной сети.

Дополнительная информация

- Dnsmasq (https://oreil.ly/vvfHg).

ГЛАВА 17

Точное время с ntpd, chrony и timesyncd

Автоматическое поддержание точного времени на вашем компьютере и на всех хостах в вашей сети легко организовать с помощью сетевого протокола времени (Network Time Protocol, NTP). Поддержка NTP в Linux реализована с помощью *ntpd* — демона NTP, *chrony* — современной замены *ntpd* и *timeyncd* в systemd. Да, все верно, есть три (как минимум) способа автоматического управления временем на вашем компьютере с Linux.

В локальных сетях ntpd и chrony могут также играть роль сервера времени, а timesyncd, как более простой и легковесный клиент, не имеет функций сервера. ntpd и chrony — полноценные реализации NTP, а timesyncd использует упрощенный протокол сетевого времени (Simple Network Time Protocol, SNTP).

Большинство дистрибутивов Linux включают настройки по умолчанию, предусматривающие использование обслуживаемых ими серверов времени. Эти серверы имеют такие имена, как 2.fedora.pool.ntp.org и 0.ubuntu.pool.ntp.org. Поэтому вам не нужно ничего делать для настройки NTP, разве только быть внимательнее и не отключить его поддержку во время установки. В данной главе вы узнаете, как проверить текущие настройки, как их изменить и как настроить сервер времени для локальной сети.

Существует Всемирная сеть серверов времени, которыми каждый желающий может пользоваться бесплатно. Сеть организована в *слои* (strata), начиная со слоя 0. Слой 0 служит источником точного времени для всего остального и включает сеть атомных часов, радиоприемников, настроенных на атомные часы, и приемников GPS, использующих сигналы, передаваемые спутниками GPS.

Следующий слой — слой 1, включающий первичные серверы времени. Первичные серверы времени в слое 1 напрямую подключены к источникам точного времени в слое 0.

Слой 2 содержит тысячи общедоступных серверов, которые синхронизируются с серверами в слое 1. Считается хорошим тоном подключаться к серверам в слое 2, чтобы предотвратить перегрузку серверов в слое 1, и не использовать серверы в слое 1 без уважительной причины.

Иерархия продолжается и ниже. Например, есть слои 4, 5 и 6 общедоступных серверов, а также частные серверы в локальных сетях, которые синхронизируются с общедоступными серверами в данных слоях. Впрочем, это не совсем так; вы можете определить свой частный сервер NTP в локальной сети как сервер в слое 10, но он не обязан подключаться к серверам уровня 9 и может задействовать любой доступный сервер времени. Вам не нужно беспокоиться о выборе правильных серверов, поскольку вы будете использовать *пулы* серверов, которые фактически являются кластерами серверов NTP, а не отдельными серверами.

Чем глубже вы будете погружаться в тему поддержки точного времени на компьютерах, тем запутаннее и непонятнее она будет казаться или, наоборот, увлекательнее, в зависимости от ваших желаний и стремлений. Посетите сайты NTP Pool Project (https://ntppool.org) и NTP: The Network Time Protocol (http://ntp.org), чтобы узнать все, что нужно, в том числе как запустить свой общедоступный сервер времени.

В вашей системе Linux есть как минимум два источника времени. Один из них — это аппаратные часы на материнской плате, которые также называются часами реального времени (Real-Time Clock, RTC). Другой — системное время, поддерживаемое ядром Linux. Часы реального времени всегда включены, даже когда компьютер выключен, и питаются от батареи или конденсатора на материнской плате. Когда система Linux загружается, настроенный вами клиент NTP получает время из RTC. Затем, когда устанавливается подключение к сети, он корректирует время, обращаясь к вышестоящему серверу времени.

Время в RTC можно установить в BIOS/UEFI, а также с помощью команд, с которыми вы познакомитесь в данной главе. Часы RTC всегда должны быть установлены в Всемирное координированное время (Coordinated Universal Time, UTC), а ядро Linux само рассчитает время для вашего часового пояса. Время UTC похоже на Среднее время по Гринвичу (Greenwich Mean Time, GMT), но не является тем же самым. UTC — это стандарт времени, а GMT — часовой пояс. Ни UTC, ни GMT не предусматривают перехода на летнее время (Daylight Saving Time, DST).

Данные о часовых поясах поступают из IETF.org Timezones (https://oreil.ly/gUnet). Это постоянно меняющиеся данные, поскольку страны то меняют даты перехода на летнее время, то отказываются от перехода на летнее время, то снова его вводят. В большинстве дистрибутивов Linux эта информация хранится в `/usr/share/zoneinfo/`. Рабочая группа инженеров Интернета (Internet Engineering Task Force, IETF) следит за этими изменениями и распространяет свои базы данных бесплатно.

17.1. Определение клиента NTP, установленного в системе Linux

Задача

Прочитав вводную часть главы, вы узнали, что синхронизацией времени в Linux управляет ntpd, chrony или timesyncd, и теперь хотите знать, какой из этих клиентов используется в вашей системе Linux.

Решение

Используйте команду `ps`, чтобы узнать, запущены ли в вашей системе какие-либо из трех демонов синхронизации времени, ntpd, chronyd или timesyncd:

```
$ ps ax|grep -w ntp
$ ps ax|grep -w chrony
$ ps ax|grep -w timesyncd
```

Если какой-то из них запущен, то переходите к соответствующим рецептам в этой главе, чтобы узнать, как управлять своим демоном времени.

Если нет, то проверьте: возможно, в вашей системе есть утилита *timedatectl*, которая является частью systemd:

```
$ timedatectl status
                      Local time: Sun 2020-10-04 10:59:48 PDT
                  Universal time: Sun 2020-10-04 17:59:48 UTC
                        RTC time: Sun 2020-10-04 17:59:48
                       Time zone: America/Los_Angeles (PDT, -0700)
       System clock synchronized: no
systemd-timesyncd.service active: no
                 RTC in local TZ: no
```

В данном случае timedatectl не обнаружила никаких демонов времени, на что указывает строка `systemd-timesyncd.service active: no`. Выполните дополнительную проверку, запросив статус службы systemd-timesyncd:

```
$ systemctl status systemd-timesyncd
● systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; disabled;
vendor preset: enabled)
   Active: inactive (dead)
     Docs: man:systemd-timesyncd.service(8)
```

Вывод показывает, что служба не запущена и, соответственно, данная система не синхронизирует время с внешним миром и определяет время только по системным часам реального времени (RTC). Если это ваш случай, то вам необходимо настроить ntpd, chrony или timesyncd.

Комментарий

Некоторые дистрибутивы Linux не используют systemd; см. рецепт 4.1, в котором рассказывается, как узнать, используется ли эта система инициализации в вашем дистрибутиве. Если вы используете дистрибутив Linux без systemd, то у вас на выбор есть только два варианта: ntpd или chrony.

Если обнаружится, что ntpd и chrony работают одновременно в одной системе, то избавьтесь от ntpd, поскольку chrony — более новая, быстрая и надежная реализация. А кроме того, выполняясь одновременно, они будут конфликтовать друг с другом.

Вывод timedatectl содержит много полезной информации. Пример, представленный выше, показывает, что часы реального времени (RTC) правильно настроены на Всемирное координированное время (UTC), а в качестве системного часового пояса настроено Тихоокеанское летнее время (Pacific Daylight Time, PDT). Служба systemd-timesyncd.service не запущена, и время в системе не синхронизировано.

Дополнительная информация

- timedatectl: инструмент управления временем и датой в системе (https://oreil.ly/QddJ7).
- `man 1 ps`

17.2. Использование timesyncd для простой синхронизации времени

Задача

Узнать, как настроить простейший клиент NTP, чтобы на компьютере всегда было установлено правильное время.

Решение

Включите синхронизацию с общедоступным NTP-сервером с помощью демона *systemd-timesyncd*, которому требуется systemd. Проверьте статус systemd-timesyncd:

```
$ systemctl status systemd-timesyncd
● systemd-timesyncd.service - Network Time Synchronization
     Loaded: loaded (/usr/lib/systemd/system/systemd-timesyncd.service;
```

```
      disabled; vendor preset: enabled)
     Active: inactive (dead)
       Docs: man:systemd-timesyncd.service(8)
```

Включите его с помощью утилиты timedatectl и убедитесь, что systemd-timesyncd запустился:

```
$ timedatectl set-ntp true
$ systemctl status systemd-timesyncd
● systemd-timesyncd.service - Network Time Synchronization
   Loaded: loaded (/lib/systemd/system/systemd-timesyncd.service; enabled;
vendor preset: enabled)
   Active: active (running) since Sun 2020-10-04 18:17:51 PDT; 16min ago
     Docs: man:systemd-timesyncd.service(8)
 Main PID: 3990 (systemd-timesyn)
   Status: "Synchronized to time server 91.189.89.198:123 (ntp.ubuntu.com)."
    Tasks: 2 (limit: 4915)
   CGroup: /system.slice/systemd-timesyncd.service
           └─3990 /lib/systemd/systemd-timesyncd

Oct 04 18:17:51 pc systemd[1]: Starting Network Time Synchronization...
Oct 04 18:17:51 pc systemd[1]: Started Network Time Synchronization.
Oct 04 18:33:01 pc systemd-timesyncd[3990]: Synchronized to time server
91.189.89.198:123 (ntp.ubuntu.com).
```

Если демон systemd-timesyncd не запустился, то запустите его:

```
$ sudo systemctl start systemd-timesyncd
```

После этого утилита timedatectl должна сообщить примерно следующее:

```
$ timedatectl status
                      Local time: Sun 2020-10-04 18:35:56 PDT
                  Universal time: Mon 2020-10-05 01:35:56 UTC
                        RTC time: Mon 2020-10-05 01:35:56
                       Time zone: America/Los_Angeles (PDT, -0700)
       System clock synchronized: yes
systemd-timesyncd.service active: yes
                 RTC in local TZ: no
```

Все выглядит правильно: система синхронизирована, все времена правильные, и служба systemd-timesyncd.service активна.

Рекомендуется настроить использование нескольких общедоступных серверов времени для резервирования. Откройте файл `/etc/systemd/timesyncd.conf` и добавьте в него дополнительные серверы NTP, раскомментировав строку `NTP` и перечислив через пробел несколько пулов общедоступных серверов:

```
[Time]
NTP=0.north-america.pool.ntp.org 1.north-america.pool.ntp.org
2.north-america.pool.ntp.org
```

```
#FallbackNTP=ntp.ubuntu.com
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Комментарий

В оригинальном файле `/etc/systemd/timesyncd.conf` закомментированные параметры описывают конфигурацию по умолчанию.

Пулы серверов очень надежны, поскольку состоят из нескольких серверов. Для лучшей производительности используйте пулы серверов пулов из вашего региона, или континентальные пулы (https://oreil.ly/iEipo), или пулы из своей страны, которые можно найти, щелкнув на ссылке для своего континента.

Ваш дистрибутив Linux может быть настроен на использование своих пулов серверов, например:

```
0.opensuse.pool.ntp.org 1.opensuse.pool.ntp.org 2.opensuse.pool.ntp.org
```

Это хорошо, и вам не стоит менять данную настройку, тем более что конфигурация по умолчанию обычно более надежна.

Дополнительная информация

- Глава 4.
- NTP Pool Project (https://ntppool.org).
- `man 5 timesyncd.conf`

17.3. Настройка времени вручную с помощью утилиты timedatectl

Задача

Вручную настроить системное время и время в часах реального времени (RTC).

Решение

Используйте утилиту *timedatectl*. Она помогает установить дату, системное время и время в RTC одной командой:

```
$ timedatectl set-time "2020-10-04 19:30:00"
Failed to set time: Automatic time synchronization is enabled
```

Установить время вручную не получится, если запущен демон *systemd-timesyncd*, поэтому его нужно остановить:

```
$ sudo systemctl stop systemd-timesyncd
```

Затем введите новые значения в формате ГГГГ-ММ-ДД ЧЧ:ММ:СС и проверьте результат:

```
$ timedatectl set-ntp false
$ timedatectl set-time "2020-10-04 19:30:00"
$ timedatectl status
                      Local time: Sun 2020-10-04 19:30:06 PDT
                  Universal time: Mon 2020-10-05 02:30:06 UTC
                        RTC time: Mon 2020-10-05 02:30:06
                       Time zone: America/Los_Angeles (PDT, -0700)
       System clock synchronized: no
systemd-timesyncd.service active: no
                 RTC in local TZ: no
```

Если после этого запустить демон systemd-timesyncd, то он затрет ваши настройки.

Комментарий

Утилита timedatectl имеет небольшой набор команд. Если вы привыкли использовать команду `date` для установки времени и других операций с датой и временем, то timedatectl может показаться вам чересчур простой. Она намеренно сделана такой простой, а для выполнения сложных задач вы все еще можете использовать `date`.

Дополнительная информация

- `man 5 timesyncd.conf`

17.4. Использование chrony в роли клиента NTP

Задача

Вам нужен полноценный клиент/сервер NTP, и вы решили узнать, как настроить *chrony* в качестве клиента NTP.

Решение

Сначала проверьте, установлен ли ntpd, и если да, то удалите его. Если у вас используется демон systemd-timesyncd, то отключите его:

```
$ sudo systemctl disable systemd-timesyncd
$ sudo systemctl stop systemd-timesyncd
```

Затем установите chrony. В большинстве дистрибутивов Linux пакет тоже называется chrony. После установки используйте команду `chronyc`, чтобы проверить состояние:

```
$ chronyc activity
200 OK
8 sources online
0 sources offline
0 sources doing burst (return to online)
0 sources doing burst (return to offline)
0 sources with unknown address
```

Отлично! Как сообщает строка `8 sources online` (8 источников подключено), демон уже работает. (Если он не запустился, то прочитайте подраздел «Комментарий» далее.) Найдите файл `chrony.conf`, который может находиться в `/etc/chrony.conf` (Fedora) или в `/etc/chrony/chrony.conf` (Ubuntu), и исследуйте настройки. Вам не нужно ничего менять, чтобы использовать chrony в роли клиента. Проверьте список серверов NTP, где вы увидите параметры `server` или `pool`. Следующий пример, полученный в системе Ubuntu, включает пулы NTP-серверов Ubuntu по умолчанию и один локальный сервер:

```
pool 0.ubuntu.pool.ntp.org iburst
pool 1.ubuntu.pool.ntp.org iburst
pool 1.ubuntu.pool.ntp.org iburst
server ntp.domain.lan iburst prefer
```

Вы можете заменить некоторые пулы серверов Ubuntu своими общедоступными пулами, чтобы повысить надежность за счет разнообразия набора пулов:

```
pool 0.ubuntu.pool.ntp.org iburst
pool 1.ubuntu.pool.ntp.org iburst
pool 0.north-america.pool.ntp.org iburst
pool 1.north-america.pool.ntp.org iburst
server ntp.domain.lan iburst prefer
```

Сохранив изменения в конфигурационном файле, перезапустите chrony.

Комментарий

Параметр `iburst` означает быструю синхронизацию после сбоев в сети, а параметр `prefer` — всегда использовать этот сервер, если он доступен.

Это действительно все, что нужно для настройки клиента. chrony — это полная реализация NTP с множеством параметров; см. полное описание в руководстве `man 5 chrony.conf`.

Управление chrony осуществляется так же, как любой другой службой, с использованием следующих команд:

- `systemctl status chrony`;
- `sudo systemctl stop chrony`;
- `sudo systemctl start chrony`;
- `sudo systemctl restart chrony`.

У chrony есть несколько преимуществ перед ntpd. Основными в роли клиента являются улучшенная обработка сбоев в сети и более быстрая повторная синхронизация при восстановлении подключения.

Дополнительная информация

- chrony (https://oreil.ly/1S41c).
- `man 5 chrony.conf`
- `man 1 chronyc`

17.5. Использование chrony в роли локального сервера времени

Задача

Настроить chrony для использования в роли сервера времени в локальной сети.

Решение

Так же как в рецепте 17.4, отключите systemd-timesyncd и удалите ntpd, если он есть в вашей системе. Затем установите пакет chrony.

Найдите файл конфигурации `/etc/chrony.conf` (Fedora, openSUSE) или `/etc/chrony/chrony.conf` (Ubuntu). В следующем примере показана базовая конфигурация:

```
pool 0.north-america.pool.ntp.org iburst
pool 1.north-america.pool.ntp.org iburst
pool 2.north-america.pool.ntp.org iburst
```

```
local stratum 10

allow 192.168.0.0/16
allow 2001:db8::/56

driftfile /var/lib/chrony/chrony.drift
maxupdateskew 100.0
rtcsync
logdir /var/log/chrony
log measurements statistics tracking
leapsectz right/UTC
makestep 1 3
```

Затем на клиентских компьютерах добавьте имя сервера в их файлы `chrony.conf`:

```
server ntp.domain.lan iburst prefer
```

Параметр `prefer` означает: всегда использовать этот сервер, если он доступен. Одна из причин создания локального сервера времени — уменьшение нагрузки на общедоступные серверы времени. С помощью параметра `prefer` можно объявить локальный сервер предпочтительным, а общедоступные серверы задействовать как резервные варианты на случай, если локальный сервер станет недоступен, например:

```
server ntp.domain.lan iburst prefer
pool 1.north-america.pool.ntp.org iburst
pool 2.north-america.pool.ntp.org iburst
```

Комментарий

`local stratum 10` настраивает chrony для продолжения работы в роли локального NTP-сервера, даже если соединение с Интернетом будет разорвано, а `stratum 10` переносит ваш сервер в самый нижний слой иерархии, где он гарантированно оказывается ниже любых внешних серверов NTP, которые вы используете. Допустимые значения: 1–15. (Используйте число, отличное от 10, если вдруг этот рецепт сделает `слой 10` чересчур популярным.)

Параметр `allow` определяет сети, которым разрешено использовать ваш NTP-сервер.

`rtcsync` сообщает chrony о необходимости синхронизации часов реального времени с системным временем.

`log` включает журналирование и определяет события, которые должны фиксироваться в журнале.

Описание других параметров вы найдете в руководстве `man 5 chrony.conf` или в файле `chrony.conf` с настройками по умолчанию, который обычно содержит подробные комментарии.

Дополнительная информация

- chrony (https://oreil.ly/1S41c).
- `man 5 chrony.conf`
- `man 1 chronyc`

17.6. Вывод статистики chrony

Задача

Узнать текущие показатели работы chrony, такие как имена вышестоящих NTP-серверов, смещение, асимметрия, имя сервера, с которым выполнялась последняя синхронизация, и другую информацию.

Решение

Используйте команду `chronyc`. Подкоманда `tracking` сообщит величину корректировки, время RTC, асимметрию и другую информацию:

```
$ chronyc tracking
Reference ID    : A29FC87B (time.cloudflare.com)
Stratum         : 4
Ref time (UTC)  : Tue Oct 06 02:20:23 2020
System time     : 0.002051390 seconds fast of NTP time
Last offset     : +0.002320110 seconds
RMS offset      : 0.017948814 seconds
Frequency       : 28.890 ppm fast
Residual freq   : +0.252 ppm
Skew            : 1.250 ppm
Root delay      : 0.069674924 seconds
Root dispersion : 0.003726898 seconds
Update interval : 838.2 seconds
Leap status     : Normal
```

Получить список серверов, используемых в текущий момент, можно следующим образом:

```
$ chronyc sources
chronyc sources
```

```
210 Number of sources = 19
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^- golem.canonical.com           2   9     0   37m    +55ms[  +58ms] +/-  209ms
^- alphyn.canonical.com          2   9     0   34m    +23ms[  +25ms] +/-  158ms
^- pugot.canonical.com           2   9     0   44m    +92ms[  +80ms] +/-  229ms
^- chilipepper.canonical.com     2   9    11    31    +48ms[  +48ms] +/-  181ms
[...]
```

Список серверов с описаниями:

```
$ chronyc sources -v
210 Number of sources = 19

  .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
 / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
| /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
||                                                 .- xxxx [ yyyy ] +/- zzzz
||      Reachability register (octal) -.           |  xxxx = adjusted offset,
||      Log2(Polling interval) --.      |          |  yyyy = measured offset,
||                                \     |          |  zzzz = estimated error.
||                                 |    |           \
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================
^- golem.canonical.com           2   9     0   46m    +67ms[  +58ms] +/-  209ms
^- alphyn.canonical.com          2   9     0   44m    +35ms[  +25ms] +/-  158ms
^* pugot.canonical.com           2   9     1   54m   +104ms[  +80ms] +/-  229ms
^- chilipepper.canonical.com     2   9    11   587    +60ms[  +48ms] +/-  181ms
^- ntp.wdc1.us.leaseweb.net      2   7     4   327    +26ms[  +15ms] +/-  198ms
^- 216.126.233.109               2   9     1   459   +106ms[  +95ms] +/-  171ms
^- 157.245.170.163               3   9     1   476  +1191us[  -10ms] +/-  145ms
```

Звездочкой отмечен сервер, с которым была выполнена последняя синхронизация.

Комментарий

chrony корректирует сетевые задержки, обрабатывает разрывы соединений, а также учитывает режимы ожидания и сна клиентских машин. Часы chrony никогда не останавливаются и синхронизируют вашу сеть, даже когда внешние серверы времени недоступны.

Дополнительная информация

- `man 1 chronyc`
- `chrony.tuxfamily.org` (https://oreil.ly/1S41c).

17.7. Использование ntpd в роли клиента NTP

Задача

Вы познакомились с chrony и timesyncd и их достоинствами, но все равно решили использовать ntpd в роли NTP-клиента.

Решение

Это вполне оправданно, поскольку ntpd продолжает активно поддерживаться и отлично справляется со своей задачей. Для начала убедитесь, что ntpd — единственный клиент NTP в вашей системе (см. рецепт 17.1). В большинстве дистрибутивов Linux ntpd устанавливается в составе пакета ntp.

Кроме того, большинство дистрибутивов устанавливают практичную конфигурацию и запускают демон ntpd после установки. Проверьте, так ли это, выполнив команду `ps`:

```
$ ps ax | grep -w ntpd
3754 ?        Ssl    0:00 /usr/sbin/ntpd -u ntp:ntp -g
```

Если демон не был запущен автоматически, то запустите его:

```
$ systemctl start ntpd
```

Пока ntpd запускается, загляните в файл конфигурации, обычно `/etc/ntp.conf`. Настройки по умолчанию подходят для большинства систем и обычно не требуют изменений. Если в вашей локальной сети есть собственный сервер NTP, то добавьте его в конфигурацию в качестве основного сервера и один пул серверов (например, Fedora Linux) в качестве запасного:

```
server ntp.domain.lan iburst prefer
pool 2.fedora.pool.ntp.org iburst
```

Впрочем, можно оставить конфигурацию по умолчанию, которая подходит для большинства ситуаций. Дистрибутивы Linux обычно поддерживают свои пулы серверов NTP и включают ссылки на них в конфигурации по умолчанию. Если вы решите заменить их или добавить несколько внешних общедоступных серверов, то загляните в раздел с описанием континентальных пулов серверов NTP (https://oreil.ly/W70Ba) или используйте пулы из своей страны, которые можно найти, щелкнув на ссылке для своего континента.

После изменения настроек в `/etc/ntp.conf` перезапустите ntpd:

```
$ systemctl restart ntpd
$ sudo /etc/init.d/ntp restart
```

Убедитесь, что он работает. В следующем примере звездочкой отмечен сервер, с которым выполняется синхронизация:

```
$ ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
 2.fedora.pool.n .POOL.          16 p    -   64    0    0.000   +0.000   0.000
*ntp.domain.lan. 172.16.16.3      2 u   34  256  203   80.324  -49.772  54.508
+138.68.46.177 ( 80.153.195.191   2 u   92  256  123   90.932  -15.534  39.947
+vps6.ctyme.com  216.218.254.202  2 u  453  256   46   69.927  -29.296  84.811
+ec2-3-217-79-24 132.163.97.6     2 u  426  256  202  165.888  -51.442  93.224
```

Комментарий

Параметр `iburst` требует от ntpd выполнить быструю синхронизацию сразу после запуска.

Параметр `prefer` означает — всегда использовать этот сервер, если он доступен.

Дополнительная информация

- `man 5 ntp.conf`
- `man 8 ntpd`
- `man 8 ntpq`
- Документация для NTP (https://oreil.ly/lpDgk).

17.8. Использование ntpd в роли сервера NTP

Задача

Узнать, как запустить сервер ntpd в своей локальной сети.

Решение

ntpd может использоваться не только в роли клиента NTP, но и в роли сервера времени, например, для локальной сети. Конфигурация остается почти без

изменений, разве что нужно добавить некоторые элементы управления доступом. Следующий пример демонстрирует полное содержимое /etc/ntp.conf:

```
driftfile /var/lib/ntp/drift

restrict default nomodify notrap nopeer noquery
restrict -6 default nomodify notrap nopeer noquery
restrict 127.0.0.1
restrict ::1

pool 0.north-america.pool.ntp.org
pool 1.north-america.pool.ntp.org
pool 2.north-america.pool.ntp.org

leapfile /usr/share/zoneinfo/leap-seconds.list

statistics clockstats loopstats peerstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable
statsdir /var/log/ntpstats/
```

Комментарий

В файле, указанном в параметре driftfile, ntpd запоминает временной сдвиг, вызванный колебаниями частоты кварцевого резонатора на материнской плате. Далее кратко описываются другие настройки в этом конфигурационном файле:

- restrict default запрещает все и разрешает только то, что явно разрешено, и устанавливает значения по умолчанию;
- nomodify не позволяет другим серверам времени вносить какие-либо изменения в вашу систему. Запросы разрешены;
- notrap отключает удаленное журналирование;
- nopeer запрещает взаимную синхронизацию одноранговых серверов, поэтому единственными серверами, которым разрешено представлять службу времени, являются указанные в директивах server и pool;
- noquery запрещает удаленные запросы и удаленное журналирование;
- restrict 127.0.0.1 и restrict ::1 означает доверие localhost;
- statistics определяет характеристики для журналирования в /var/log/ntpstats/. В большинстве случаев в этом нет необходимости, но кому-то может быть интересно понаблюдать за тем, какие вышестоящие серверы NTP показывают лучшую производительность.

Дополнительная информация

- `man 5 ntp.conf`
- `man 8 ntpd`
- `man 8 ntpq`
- Документация для NTP (https://oreil.ly/lpDgk).

17.9. Управление часовыми поясами с помощью утилиты timedatectl

Задача

Перечислить все часовые пояса, узнать текущий используемый часовой пояс и изменить его.

Решение

Используйте утилиту timedatectl. Узнать текущий используемый часовой пояс можно следующим образом:

```
$ timedatectl | grep -i "time zone"
     Time zone: America/Los_Angeles (PDT, -0700)
```

Список всех часовых поясов:

```
$ timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
[...]
```

Список имеет длину более 400 строк. Используйте команду `grep`, чтобы оставить в списке только то, что вы ищете. Список включает названия столиц, поэтому его можно отфильтровать по их названиям, например Berlin:

```
$ timedatectl list-timezones | grep -i berlin
Europe/Berlin
```

Установка этого часового пояса в качестве текущего:

```
$ sudo timedatectl set-timezone Europe/Berlin
```

Изменение вступает в действие немедленно. Запустите утилиту timedatectl снова для проверки.

Комментарий

Взаимодействуя с людьми, живущими в разных часовых поясах, используйте время UTC для согласования времени встреч. Существует несколько онлайн-конвертеров часовых поясов, например Time Zone Converter (https://oreil.ly/NyLj7).

Чтобы установить текущий часовой пояс, его следует указывать в формате *«регион/город»*, как его отображает команда `timedatectl list-timezones`. Этот формат определяется стандартом ISO 8601, задающим однозначную форму выражения часовых поясов, времени и дат. Стандарт использует «нисходящую нотацию», согласно которой значения перечисляются от больших к меньшим. Например, для часовых поясов используется порядок *«континент/страна/город»*. Традиция в США записывать дату как *год-день-месяц* не соответствует этому стандарту. Стандартной для дат считается нотация *год-месяц-день* с четырехзначным годом: ГГГГ-ММ-ДД — и для времени: ЧЧ:ММ:СС в 24-часовом формате.

Официально опубликованный стандарт ISO 8601 стоит денег, но, поискав в Интернете, можно найти информацию бесплатно.

Дополнительная информация

- `man 1 timedatectl`

17.10. Управление часовыми поясами без утилиты timedatectl

Задача

Система Linux не имеет systemd, и требуется узнать, какие команды использовать для управления часовыми поясами.

Решение

Чтобы узнать текущий часовой пояс, запустите команду `date`:

```
$ date
Wed Oct  7 08:32:40 PDT 2020
```

Или загляните в файл `/etc/localtime`, который является символической ссылкой:

```
$ ls -l /etc/localtime
lrwxrwxrwx 1 root root 41 Oct  7 08:06 /etc/localtime ->
../usr/share/zoneinfo/America/Los_Angeles
```

Каталог `/usr/share/zoneinfo` содержит все часовые пояса:

```
$ ls /usr/share/zoneinfo
total 324
drwxr-xr-x  2 root root  4096 May 21 23:02 Africa
drwxr-xr-x  6 root root 20480 May 21 23:02 America
drwxr-xr-x  2 root root  4096 May 21 23:02 Antarctica
drwxr-xr-x  2 root root  4096 May 21 23:02 Arctic
[...]
```

Найдите в подкаталогах ближайший к вам город, например Мадрид:

```
$ ls /usr/share/zoneinfo/Europe
[...]
-rw-r--r-- 1 root root 2637 May  7 17:01 Madrid
-rw-r--r-- 1 root root 2629 May  7 17:01 Malta
lrwxrwxrwx 1 root root    8 May  7 17:01 Mariehamn
-rw-r--r-- 1 root root 1370 May  7 17:01 Minsk
[...]
```

Измените свой часовой пояс, заменив ссылку `/etc/localtime`:

```
$ sudo ln -sf /usr/share/zoneinfo/Europe/Madrid/etc/localtime
```

Изменение вступает в силу немедленно.

Комментарий

Ниже представлена интересная команда, которая перечислит все часовые пояса в алфавитном порядке:

```
$ php -r 'print_r(timezone_identifiers_list());'
Array
(
    [0] => Africa/Abidjan
    [1] => Africa/Accra
    [2] => Africa/Addis_Ababa
    [3] => Africa/Algiers
    [4] => Africa/Asmara
[...]
```

Команда `php` устанавливается с пакетом *php-cli*.

Графические среды рабочего стола также имеют удобные утилиты с графическим интерфейсом для управления временем, датой и часовыми поясами. Если на вашем рабочем столе отображаются часы, то попробуйте щелкнуть на них правой кнопкой мыши, чтобы открыть панель свойств или настроек.

Дополнительная информация

- `man 1 date`
- `man 1 ln`

ГЛАВА 18

Создание брандмауэра/маршрутизатора для подключения к Интернету на Raspberry Pi

Одноплатный компьютер Raspberry Pi (RPi) прекрасно подходит на роль брандмауэра/маршрутизатора для подключения к Интернету небольших сетей и стоит совсем недорого. Для этой цели подойдет любая модель Raspberry Pi, но я рекомендую Raspberry Pi 4 Model B, как более мощную и первую в семействе Pi с гигабитным портом Ethernet.

Обзор

В этой главе вы узнаете, как установить ОС Raspberry Pi, подключить Pi к монитору компьютера или телевизору, использовать режим восстановления ОС Raspberry Pi, запустить Pi без монитора, добавить второй порт Ethernet, организовать общий доступ к Интернету и использовать Pi для реализации службы имен в локальной сети.



Во всех примерах в этой главе используется модель Raspberry Pi 4 Model B с операционной системой Raspberry Pi. Раньше данная операционная система называлась Raspbian. В ее основе лежит Debian Linux, поэтому пользователи Debian, Ubuntu, Mint или любого другого варианта будут чувствовать себя в ней как дома.

Плюсы и минусы брандмауэра/маршрутизатора на Raspberry Pi

Raspberry Pi — это универсальный компьютер, а не специализированный брандмауэр/маршрутизатор. У него есть интерфейсы Wi-Fi, Ethernet и Bluetooth, и он работает под управлением Linux. Для сравнения, обычно для небольших сетей выбираются небольшие устройства, сочетающие в себе брандмауэр, маршрутизатор, точку беспроводного доступа и коммутатор Ethernet, такие как Linksys AC1900 или TP-Link Archer AX20. Они имеют интерфейс Wi-Fi, гигабитный Ethernet, несколько антенн и поддержку «умных» сервисов, таких как «Алекса», а также приложения для администрирования со смартфона.

Недостатком этих устройств является их негибкость, ограниченные объем памяти и поддержка операционных систем. Если вы решите заменить программное обеспечение производителя, то вам придется использовать специализированные дистрибутивы для маршрутизаторов, такие как OpenWRT, DD-WRT, pfSense или OPNsense. Все они превосходны, но процедура замены очень сложна и вам придется найти систему, поддерживающую ваше устройство.

Ниже представлены некоторые из преимуществ Raspberry Pi по сравнению с такими устройствами:

- гибкость, подобно любому универсальному компьютеру с Linux;
- больше памяти и места для хранения;
- подключается к монитору компьютера или телевизору, клавиатуре и мыши;
- работает под управлением множества дистрибутивов Linux, что позволит вам использовать существующие знания и не изучать какой-то новый странный интерфейс или набор команд;
- работает под управлением некоторых других операционных систем, таких как *bsd, Windows 10, Android, Chromium и др.;
- может связываться с мобильной точкой доступа;
- 64-разрядная поддержка.

Raspberry Pi можно запустить с графическим рабочим столом и без графического рабочего стола с подключением через SSH, подобно любой системе Linux.

Недостатки Raspberry Pi:

- не может работать как коммутатор Ethernet, подобно специализированным устройствам;
- интерфейс Wi-Fi не такой мощный, как у специализированных устройств;

- модели Raspberry Pi 3 и старше имеют низкую производительность Ethernet, поскольку порт Ethernet использует шину USB совместно с другими устройствами; в RPi 4 эта проблема решена за счет выделенного порта Gigabit Ethernet.

Для Raspberry Pi существует несколько специализированных дистрибутивов Linux. Официальная операционная система называется Raspberry Pi OS. Она основана на Debian Linux. Свои варианты для Raspberry Pi имеют SUSE, Ubuntu, Fedora, Arch Linux ARM и MX Linux. Существуют также специализированные медиасерверы и игровые дистрибутивы. В данной главе мы остановимся на Raspberry Pi OS, поскольку она оптимизирована для Pi и предлагает полноценный графический рабочий стол с приличной производительностью даже на старых моделях RPi. Операционная система Raspberry Pi OS мало отличается от любого другого дистрибутива Linux, поэтому все, что доступно на большом Linux-компьютере, будет доступно и на Raspberry Pi.

Аппаратная архитектура

В основе RPi находится однокристалллная система Broadcom SoC, в которой центральный процессор, графический процессор и схемы ввода/вывода находятся на одном кристалле. Особенно впечатляет Raspberry Pi 4 Model B, поддерживающая работу с двумя экранами одновременно. Эта модель прекрасно справляется с воспроизведением фильмов высокой четкости.

Broadcom SoC — это архитектура ARM, а не x86_64, доминирующая на рынке персональных компьютеров. Процессоры ARM менее сложные и используют сокращенный набор команд (Reduced Instruction Set Computer, RISC). Процессоры x86_64 — это компьютеры со сложным набором команд (Complex Instruction Set Computer, CISC). Процессоры x86_64 намного сложнее и потребляют больше энергии.

Многообразие продуктов Raspberry Pi

Каждая когда-либо созданная модель Raspberry Pi по-прежнему доступна, включая обновленные версии Raspberry Pi 1, модели A и B. Модели A — это более дешевые версии каждого выпуска, а модели B имеют дополнительные возможности, но стоят немного дороже.

На выбор имеется также множество других вариантов Pi, например:

- Raspberry Pi Zero, самый маленький компьютер Pi за 5 долларов США;
- Raspberry Pi Zero W включает интерфейс Wi-Fi и стоит 10 долларов США;

- комплект для сборки персонального компьютера Raspberry Pi 400 — полноценная система, интегрированная в компактную клавиатуру (вам останется только добавить дисплей), за 100 долларов США.

Цены на RPi оставались стабильными с момента выпуска первой модели RPi, хотя, конечно, ситуация может измениться.

Имеется также богатый выбор аксессуаров: корпуса, сенсорные экраны, радиаторы, вентиляторы, коммутационные платы, платы расширения, всевозможные кабели и адаптеры, моторы, камеры, аудиокарты, маленькие беспроводные клавиатуры с сенсорными панелями, игровые эмуляторы, RGB-матрицы, USB-концентраторы, часы реального времени, крошечные дисплеи... Это один из лучших конструкторов на свете.

История появления и назначение

Raspberry Pi — настоящее чудо. Первоначально создатель этой платформы, Эбен Аптон (Eben Upton), намеревался получить небольшой и недорогой компьютер, чтобы побудить молодых студентов к изучению информатики, особенно студентов, которые не могли позволить себе купить полноценный персональный компьютер. Первый Raspberry Pi Version 1 Model B стоил около 35 долларов. Добавьте клавиатуру и мышь, подключитесь к телевизору или монитору, и менее чем за десятую часть цены обычного ПК вы получите рабочий компьютер с Linux, поддерживающий аудио, видео, Ethernet и USB. Открытая архитектура в сочетании с открытым программным обеспечением способствует изучению и обучению.



Микросхемы Broadcom, на которых работает Pi, не являются открытыми. Это обстоятельство остается камнем преткновения с момента выпуска первого Pi. Схемы открыты и доступны на RaspberryPi.org, операционная система и BIOS тоже открыты. По моему скромному мнению, платформа с полностью открытым исходным кодом, работающая здесь и сейчас, намного лучше, чем ничего.

Первый выпуск Raspberry Pi немедленно обрел большую популярность: за первые шесть месяцев после выхода в феврале 2012 года было продано более 500 000 единиц. С тех пор было продано около 30 миллионов Raspberry Pi. Текущая версия, Version 4 Model B, претерпела значительное обновление и является самой мощной моделью на данный момент. Она имеет:

- два порта USB 2;
- два порта USB 3;

- два порта micro HDMI с поддержкой двух дисплеев 4K;
- один выделенный гигабитный порт Ethernet;
- поддерживается объем ОЗУ от 2 до 8 Гбайт;
- 64-разрядный процессор Broadcom BCM2711 с четырьмя ядрами Cortex-A72 (ARM v8) на одном кристалле с тактовой частотой 1.5 ГГц;
- интерфейс Wi-Fi 2.4 GHz и 5.0 GHz IEEE 802.11ac;
- Bluetooth;
- 40-контактный разъем универсального интерфейса ввода/вывода (GPIO).

По сравнению с современными процессорами Intel и AMD, эти характеристики совсем не впечатляют, но такой мощности вполне достаточно, чтобы запустить графический рабочий стол на Linux, воспроизводить музыку, фильмы, просматривать веб-страницы, писать документы... — довольно неплохо для такого размера и цены.

Raspberry Pi разработан и производится зарегистрированной некоммерческой благотворительной организацией Raspberry Pi Foundation. Она поддерживает многочисленные образовательные проекты для учителей и студентов; посетите сайт организации https://raspberrypi.org, на котором можно найти текущие учебные материалы и самую свежую информацию.

18.1. Включение и выключение Raspberry Pi

Задача

На плате компьютера Raspberry Pi (RPi) нет выключателя питания, и хотелось бы узнать, как включать и выключать его.

Решение

Чтобы включить его, подключите разъем питания. Чтобы выключить — выберите в меню вашей операционной системы пункт, управляющий выключением, а затем отключите компьютер от электросети.

Комментарий

Чтобы выключить RPi, нужно отключить его от сети, а чтобы снова включить — подключить разъем питания.

Меня не удивит, если где-то продаются выключатели питания для RPi, но я их не видела. Единственная альтернатива — подключить к электросети через удлинитель с выключателем.

Дополнительная информация

- https://raspberrypi.org.

18.2. Поиск дополнительного оборудования и руководств

Задача

Вы купили Raspberry Pi 4 Model B и теперь хотите узнать, какие еще устройства нужно приобрести, чтобы пользоваться им.

Решение

Если допустить, что у вас уже есть монитор или телевизор, мышь и клавиатура, то вам также понадобятся:

- блок питания для Raspberry Pi;
- кабель HDMI — micro HDMI;
- карта micro SD емкостью не меньше 16 Гбайт;
- охлаждающий вентилятор или радиатор для процессора, модуль ОЗУ и контроллер USB;
- корпус;
- устройство для чтения карт micro SD.

Для начала запустите установку Raspberry Pi OS на карту micro SD на другом компьютере. Пока она выполняется, соберите свое оборудование. Когда все будет готово, подключите блок питания и посмотрите, как загружается новая система. (См. рецепты 18.4 и 18.5, чтобы узнать, как установить Raspberry Pi OS.)

В Интернете существует множество источников информации о RPi: схем, спецификаций, практических рекомендаций и интересных идей. В разделе «Дополнительная информация» этого рецепта представлен хороший список ресурсов, с которого можно начать.

Комментарий

Если на вашем дисплее нет порта HDMI, то см. рецепт 18.6.

Вы можете использовать любую карту micro SD с объемом не менее 16 Гбайт. Высокоскоростные карты дают заметное улучшение производительности. 16 Гбайт — это довольно мало, и желательно использовать карту побольше.

RPi 4B поддерживает три конфигурации оперативной памяти: 2, 4 и 8 Гбайт. Возможность модернизации отсутствует, поэтому что вы купите, то и будете иметь. Двух гигабайт вполне достаточно для интернет-шлюза и использования Raspberry Pi в качестве легковесного рабочего стола Linux. Для обработки мультимедиа, компиляции кода, игр и других задач необходимо больше памяти.

RPi 4B существенно мощнее и греется сильнее старых версий Pi. См. рецепт 18.3, где описывается, как обеспечить дополнительное охлаждение.

Благодаря четырем портам USB, RPi 4B обладает большой гибкостью. Вы можете использовать стандартные USB-клавиатуры и мыши, клавиатуру с трекпадом и переходники USB-PS/2 для подключения старых клавиатур и мышей. Можно подключать и жесткие диски USB, чтобы получить дополнительный объем для хранения или резервного копирования, а также любые другие USB-устройства, как и на большом компьютере. К 40-контактному разъему GPIO можно подключить практически любую плату расширения.

Существует множество хороших наборов, в которых есть все необходимое для начала работы. Мой любимый интернет-магазин — Adafruit.com (https://adafruit.com), и на сайте https://raspberrypi.org можно найти множество ссылок на другие магазины.

Дополнительная информация

На следующих сайтах можно найти превосходные руководства по Raspberry Pi:

- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2);
- Adafruit (https://oreil.ly/5qwn5);
- MagPi (https://oreil.ly/EuMY7);
- Hackspace (https://oreil.ly/5KY5B);
- Maker Pro (https://oreil.ly/NQcB0);
- Makezine (https://oreil.ly/foZqS).

18.3. Охлаждение Raspberry Pi

Задача

Процессор на плате Raspberry Pi горячий на ощупь, и вам хотелось бы организовать его охлаждение.

Решение

Установите охлаждающий вентилятор в корпус или радиаторы на процессор, модуль ОЗУ и контроллер USB. Вообще Raspberry Pi 4 греется сильнее, чем старые версии Pi, поэтому желательно установить и радиаторы, и вентилятор.

Выполните встроенную команду `vcgencmd` для измерения температуры процессора до и после установки охлаждающих устройств, а также до и после выполнения ресурсоемких задач, например компиляции кода или воспроизведения видео. В следующем примере показаны результаты замера температуры платы без вентилятора в режиме ожидания со снятой крышкой корпуса, а затем через пять минут после начала воспроизведения фильма в разрешении 1080p:

```
$ vcgencmd measure_temp
temp=48.3'C

$ vcgencmd measure_temp
temp=61.9'C
```

А вот какую температуру имеет процессор после установки охлаждающего вентилятора при воспроизведении того же фильма:

```
$ vcgencmd measure_temp
temp=52.1'C
```

Температура не должна превышать 70°C. От 40 до 60°C — это нормальный рабочий диапазон.

Дополнительная информация

- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2).
- Adafruit (https://oreil.ly/5qwn5).
- MagPi (https://oreil.ly/EuMY7).

18.4. Установка Raspberry Pi OS с помощью Imager и команды dd

Задача

Вы купили все необходимое и теперь хотите установить операционную систему.

Решение

Создайте загрузочную карту micro SD на другом компьютере, затем установите карту в Raspberry Pi и включите его.

Есть четыре способа получить загрузочную SD-карту:

- использовать программу Raspberry Pi Imager (в настоящее время доступна только для систем `.deb`, таких как Debian, Ubuntu или Mint);
- использовать установщик NOOBS (см. рецепт 18.5);
- скопировать установочный образ на карту micro SD с помощью команды `dd`;
- купить карту micro SD с уже загруженным установщиком.

Я предпочитаю вариант с NOOBS, поскольку он доступен во всех дистрибутивах Linux и добавляет режим аварийной загрузки.

Чтобы установить Raspberry Pi Imager из пакета `.deb`, скачайте его с сайта https://raspberrypi.org. Затем установите пакет:

```
$ sudo dpkg -i imager_1.5_amd64.deb
```

Пользователи Ubuntu могут установить Raspberry Pi Imager с помощью диспетчера пакетов *apt*:

```
$ sudo apt install rpi-imager
```

Вставьте SD-карту в компьютер. Найдите ее с помощью команды `lsblk -p` (см. рецепт 10.9).

Запустите Raspberry Pi Imager из системного меню и полюбуйтесь на его веселый малиновый логотип. Нажмите кнопку Operating System (Операционная система) и выберите операционную систему, которую хотите установить, и Imager скачает ее и скопирует на вашу SD-карту. Если вы уже скачали образ, то прокрутите меню выбора до Use Custom (Использовать свой), чтобы выбрать свой образ. Вы увидите диалог, подобный изображенному на рис. 18.1.

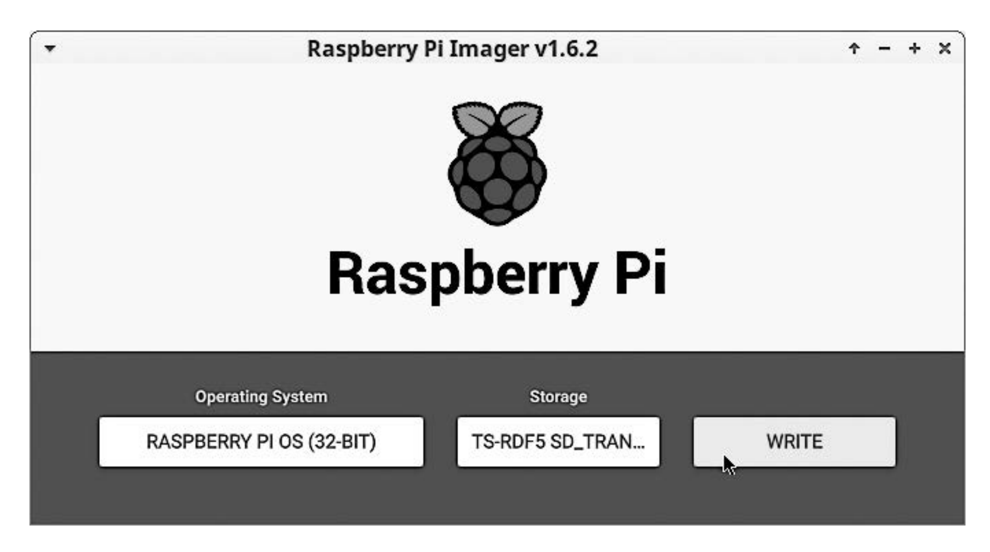


Рис. 18.1. Создание загружаемой карты micro SD с помощью Raspberry Pi Imager

Нажмите кнопку SD Card (Карта SD), чтобы выбрать устройство, а затем кнопку Write (Записать), чтобы установить операционную систему.

Если вы пользуетесь другой системой, отличной от Ubuntu, то скачайте выбранный образ операционной системы с сайта https://raspberrypi.org и скопируйте его на SD-карту с помощью команды `dd`. Ниже показан пример распаковывания и копирования образа `2021-03-24-raspios-buster-armhf.zip` на SD-карту:

```
$ sudo unzip -p 2021-03-24-raspios-buster-armhf.zip | \
  sudo dd of=/dev/foo bs=4M conv=fsync status=progress
```

Когда SD-карта будет готова, вставьте ее в Raspberry Pi и включите. После загрузки вам останется выполнить некоторые заключительные настройки, и затем ваш компьютер будет готов к использованию.

Пользователь по умолчанию — pi. Поищите в `/home/pi/Bookshelf` копию официального руководства Гарета Халфакри (Gareth Halfacree) *The Official Raspberry Pi Beginner's Guide* (Raspberry Pi Press) в формате PDF.

Комментарий

Перед копированием образа с помощью команды `dd` нет необходимости форматировать SD-карту.

Imager создаст два раздела: раздел FAT32/boot размером 256 Мбайт и корневой раздел с Ext4 достаточно большого размера для размещения файловой системы. В моей тестовой системе размер этого раздела составил 3,4 Гбайт, остальная часть карты осталась нераспределенной.

При первой загрузке корневая файловая система развернется и заполнит все нераспределенное пространство. Вы можете сжать корневую файловую систему и создать дополнительные разделы с помощью GParted (см. главу 9) или `parted` (см. главу 8).

Дополнительная информация

- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2).

18.5. Установка Raspberry Pi методом NOOBS

Задача

Установить Raspberry Pi OS с помощью NOOBS.

Решение

NOOBS (new out of the box software — «новое программное обеспечение из коробки») — это более старый установщик. Он работает во всех дистрибутивах Linux и добавляет поддержку режима загрузки для восстановления, чего не делает Raspberry Pi Imager.

Скачайте NOOBS на другой компьютер, распакуйте архив, скопируйте все его файлы на карту micro SD, затем вставьте SD-карту в свой Raspberry Pi и запустите его.

Скачать NOOBS можно на сайте https://raspberrypi.org. Существует две версии: NOOBS и NOOBS Lite. NOOBS включает Raspberry Pi OS и сетевой установщик для других операционных систем. NOOBS Lite включает только сетевой установщик.

Распакуйте NOOBS после скачивания:

```
$ unzip NOOBS_lite_v3_5.zip
```

Вставьте SD-карту в компьютер. Найдите ее с помощью команды `lsblk -p` (см. рецепт 10.9).

Создайте на карте один раздел, занимающий весь объем, и отформатируйте его в файловую систему FAT32.

Скопируйте все файлы NOOBS на SD-карту, а затем загрузите с нее Raspberry Pi. Сначала вы увидите экран, расцвеченный всеми цветами радуги, затем откроется меню установки (рис. 18.2). Настройте сеть, чтобы использовать сетевой установщик или получать обновления после установки. Выберите операционную систему, а затем займитесь чем-нибудь другим, пока не закончится процесс установки.

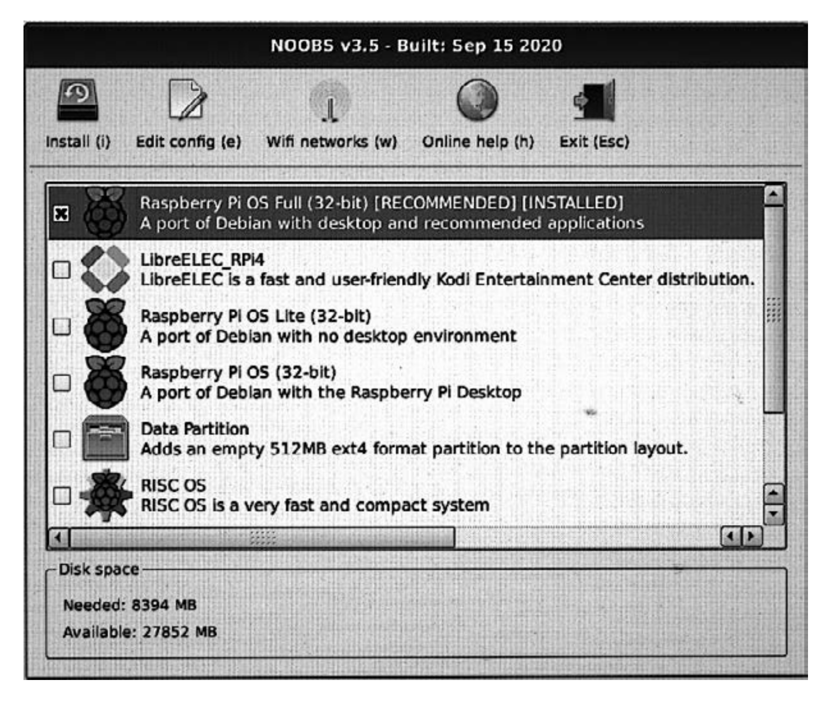


Рис. 18.2. Экран установки NOOBS

По окончании установки вам будет предложено выполнить некоторые заключительные настройки, после чего Raspberry Pi будет готов к использованию.

Пользователь по умолчанию — pi. Поищите в `/home/pi/Bookshelf` копию официального руководства Гарета Халфакри (Gareth Halfacree) *The Official Raspberry Pi Beginner's Guide* (Raspberry Pi Press) в формате PDF.

Комментарий

Способ с использованием NOOBS, по сути, заключается в распаковке и копировании, поэтому работает на любом компьютере.

Копирование файлов NOOBS на SD-карту может занять много времени. Обычно быстрее скопировать ZIP-файл на SD-карту, а затем распаковать его там. После установки ZIP-файл можно удалить.

Дополнительная информация

- Глава 9.
- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2).

18.6. Подключение дисплея без HDMI

Задача

Имеется телевизор или монитор без порта HDMI, и к нему нужно подключить Raspberry Pi.

Решение

Есть четыре способа подключить монитор к RPi 4B. Для этого можно использовать:

- маленький дисплей, выпускаемый специально для Raspberry Pi;
- адаптер DVI — HDMI;
- адаптер VGA — HDMI;
- композитное видео RCA (с кабелями для Raspberry Pi).

Для RPi выпускается множество разных дисплеев: сенсорные, LED, LCD, OLED, eInk — самые разные. Подключая такой дисплей, следуйте прилагаемой инструкции по установке.

Адаптеры DVI — HDMI и VGA — HDMI подключаются к монитору, а затем к адаптеру подключается кабель HDMI — micro-HDMI. При подключении одного дисплея HDMI к RPi 4B подключайте его к порту HDMI 0, который находится ближе всего к порту питания.

Композитное видео требует выполнения дополнительных действий на RPi 4B, поскольку по умолчанию его поддержка выключена. Самый простой способ — найти монитор с HDMI для использования при первой загрузке Pi и завершить установку. После установки откройте инструмент настройки и включите поддержку композитного видео:

```
$ sudo raspi-config
```

Выберите в списке пункт 6 Advanced Options, затем выберите A8 HDMI/Composite, далее выберите V2 Enable Composite. Выйдите из `raspi-config`, выключите Pi, подключите дисплей к композитному выходу Pi и снова включите Pi. По умолчанию изображение должно выводиться на монитор через композитный видеовыход.

Комментарий

Старые плоские мониторы обычно имеют разъемы VGA и композитного видео, поэтому подключить их можно либо через композитный разъем, либо через адаптер VGA — HDMI. Технически HDMI обеспечивает более высокое качество изображения, чем композитное видео, но большинство людей не заметят разницы.

На рис. 18.3 показан адаптер Rocketfish DVI — HDMI, кабели композитного видео, Rasberry Pi 4B в корпусе CanaKit со снятой крышкой и карта micro SD.

Комплект кабелей композитного аудио/видео-RCA с желтым, красным и белым штекерами подключается к маленькому 3,5-миллиметровому TRRS-порту на RPi. Вам нужен комплект кабелей, специально выпускающихся для RPi, поскольку нет четкого стандарта на расключение штекера TRRS. Вы можете обнаружить, что некоторые кабели распаяны не так, как предполагалось: желтый — для видеосигнала, а красный и белый — для аудиосигнала. Не используйте композитные кабели для видеокамер и MP3-плееров с не совсем обычным штекером TRRS, у которого заземляющее кольцо находится не в том месте. Штекер TRRS (tip-ring-ring-sleeve — «наконечник-кольцо-кольцо-втулка») должен быть распаян следующим образом:

```
Наконечник     Кольцо 1        Кольцо 2  Втулка
Левый динамик  Правый динамик  Земля     Видео
```

Есть несколько вариантов настройки параметров композитного видео в файле `/boot/config.txt`. Если вы планируете использовать композитное видео, то

воспользуйтесь установщиком NOOBS. В этом случае, если вам понадобится исправить его настройки, вы сможете загрузиться в режиме восстановления (см. рецепт 18.7) и получить доступ к `/boot/config.txt`.

Параметр `sdtv_mode=` задает стандарт ТВ. `sdtv_mode=0` — режим по умолчанию для Северной Америки.

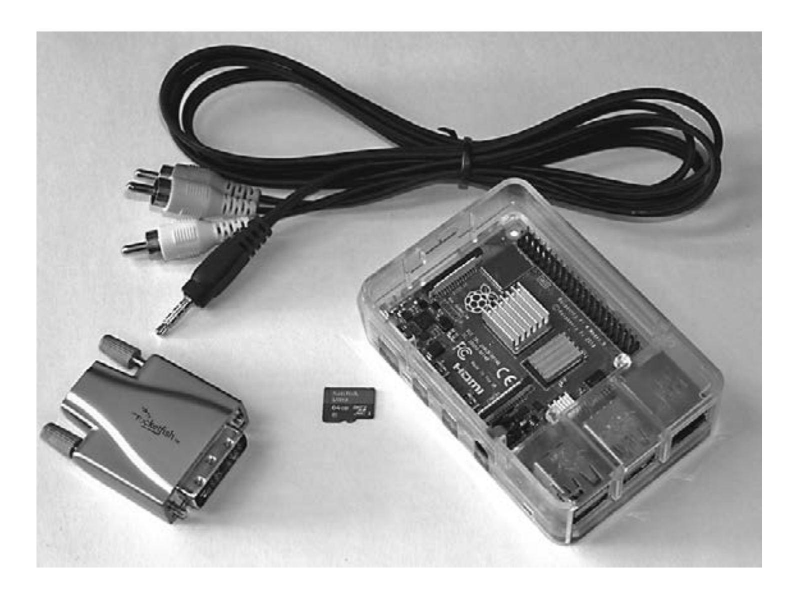


Рис. 18.3. Raspberry Pi 4B с аксессуарами

Большая часть мира использует стандарт PAL; значения режимов и соответствующие им стандарты приводятся в табл. 18.1.

Таблица 18.1. Значения для параметра sdtv_mode

Значение	Режим
0	Обычный NTSC (по умолчанию)
1	Японская версия NTSC
2	Обычный PAL
3	Бразильская версия PAL
16	NTSC с прогрессивной разверткой
18	PAL с прогрессивной разверткой

Параметр `sdtv_aspect=` определяет отношение сторон (табл. 18.2).

Таблица 18.2. Значения для параметра sdtv_aspect, определяющего отношение сторон монитора

Значение	Отношение
1	4:3 (по умолчанию)
2	14:9
3	16:9

Дополнительная информация

- Параметры настройки видео в файле `config.txt` (https://oreil.ly/yp7Mu).

18.7. Загрузка в режиме восстановления

Задача

Узнать, как загрузиться в режиме восстановления, если возникнут проблемы.

Решение

Для этого операционная система должна быть установлена с помощью NOOBS, поскольку это единственный установщик, который добавляет поддержку режима восстановления. Включите Pi и смотрите на экран. На короткое время появится экран с логотипом Raspberry Pi и сообщением For recovery mode, hold Shift (Для входа в режим восстановления удерживайте нажатой клавишу Shift). Нажмите и удерживайте клавишу Shift, пока не появится экран восстановления (экран восстановления выглядит так же, как экран установки NOOBS; см. рис. 18.2).

Экран восстановления отображает красивый графический интерфейс, позволяющий выполнять некоторые простые операции. Вы можете подключиться к Интернету, просмотреть интерактивную справку, отредактировать файл `/boot/config.txt` или отказаться от установки и установить что-то другое.

Комментарий

Экран восстановления выглядит так же, как экран установки NOOBS. Однако он доступен, только если установка Raspberry Pi OS была выполнена с помощью NOOBS, хотя к тому времени, когда вы будете читать эти строки, все может быть иначе.

Дополнительная информация

- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2).

18.8. Добавление второго интерфейса Ethernet

Задача

Вы предполагали использовать свой Raspberry Pi в качестве брандмауэра/маршрутизатора для выхода в Интернет, но у него только один порт Ethernet, а вам нужны два.

Решение

Есть два способа добавить второй порт Ethernet: с помощью адаптера USB — Ethernet или путем установки порта Ethernet, который подключается к контактам универсального разъема GPIO.

USB — Ethernet — это просто; нужно лишь подключить его.

Установка дополнительного проводного порта Ethernet требует немного больше работы. Вам понадобится адаптер Ethernet на основе контроллера Ethernet ENC28J60 (рис. 18.4).

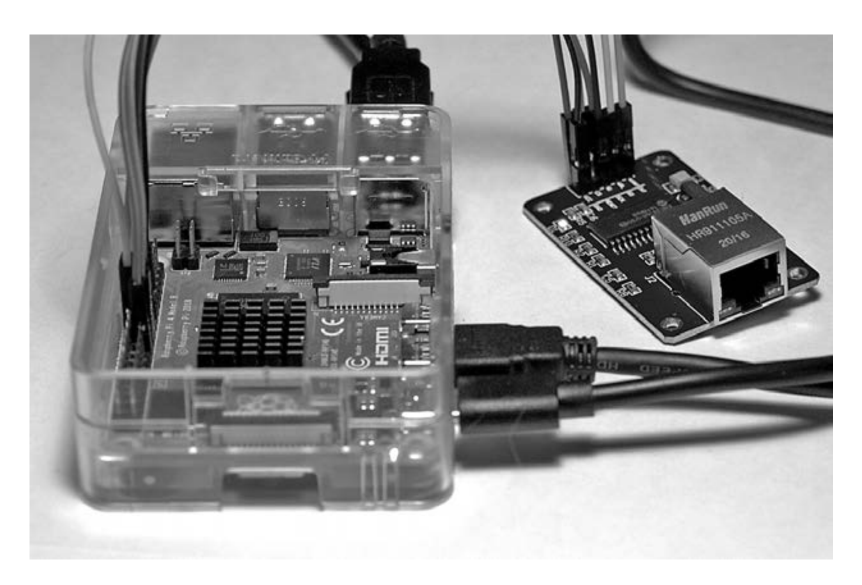


Рис. 18.4. Адаптер Ethernet ENC28J60

Чтобы подключить адаптер HanRun HR911105A, изображенный на рис. 18.4, потребуется семь одноконтактных разъемов для подключения адаптера к GPIO. К счастью, такие разъемы и наборы разноцветных проводов широко доступны и стоят недорого.

Перед подключением проводов выведите на экран схему распиновки, выполнив команду `pinout` на Raspberry Pi (рис. 18.5).

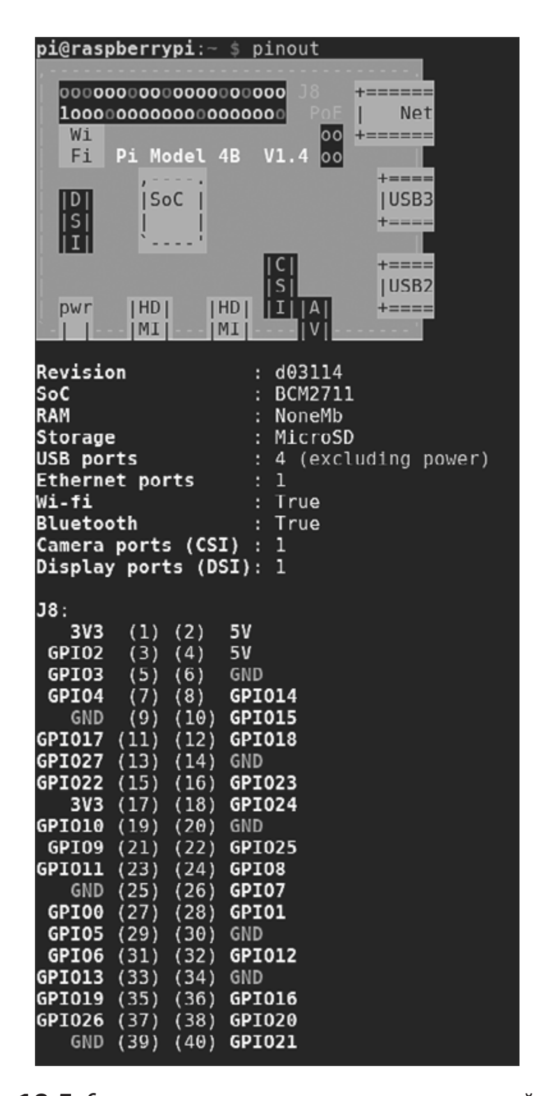


Рис. 18.5. Схема распиновки, сгенерированная командой pinout

На рис. 18.6 показана схема с номерами, опубликованная на RaspberryPi.org, с обозначениями контактов GPIO.

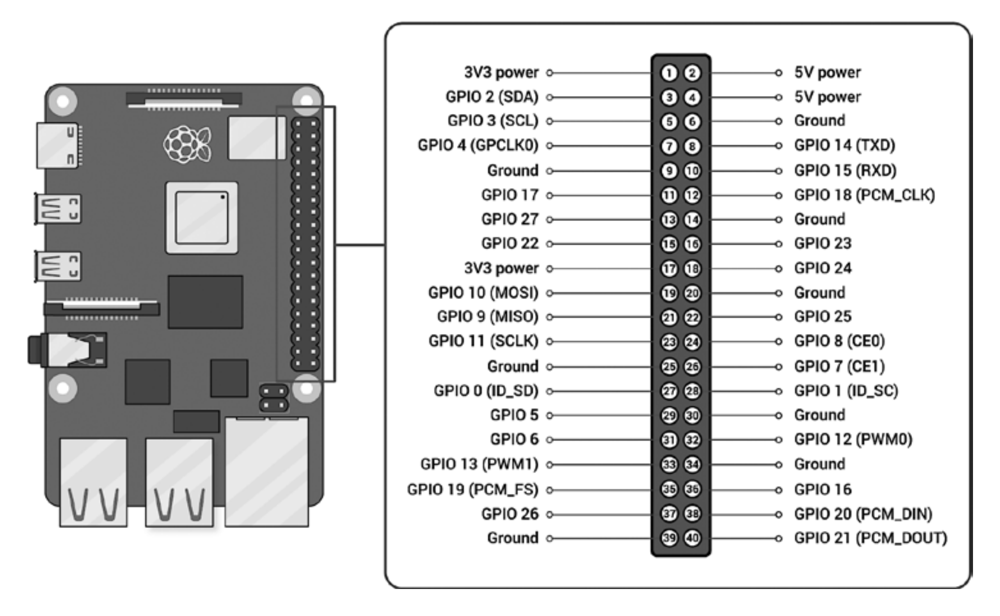


Рис. 18.6. Схема с обозначениями контактов с сайта RaspberryPi.org

Отредактируйте файл `/boot/config.txt`, чтобы включить новый порт Ethernet и загрузить драйверы:

```
dtparam=spi=on
dtoverlay=enc28j60
```

Сохраните копию схемы распиновки и выключите Raspberry Pi. Соедините проводами следующие контакты на плате RPi и ENC28J60:

```
RPi           ENC28J60
----------------------------
+3V3          VCC
GPIO10        SI
GPIO9         SO
GPIO11        SCK
GND           GND
GPIO25        INT
CE0#/GPIO8    CS
```

Обратите внимание на ориентацию контактов GPIO: контакт 1 находится на конце платы RPi рядом со слотом для SD-карты. Начните с контакта 17 (3V3), тогда все ваши провода окажутся по одну сторону, а другой контакт 3V3 останется свободен для питания вентилятора корпуса.

Подсоединив все провода, включите RPi. Запустите команду `ifconfig`, чтобы увидеть новый интерфейс Ethernet. Он должен получить имя `eth1`:

```
$ ip link show dev eth1
2: eth1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel
state DOWN mode DEFAULT group default qlen 1000
    link/ether d0:50:99:82:e7:2b brd ff:ff:ff:ff:ff:ff
```

Теперь можно приступать к настройке.

Комментарий

Контроллер Ethernet ENC28J60 поддерживает скорость передачи только до 10 Мбит/с. Если ваше подключение к Интернету имеет скорость не выше 10 Мбит/с, то возможностей этого контроллера будет вполне достаточно. С адаптером USB 3.0 Ethernet на Raspberry Pi 4 можно получить до 900 Мбит/с.

Старые модели Raspberry Pi имеют более медленный порт Ethernet, поскольку он работает на шине USB 2.0. RPi 4B — первая модель в семействе Pi с выделенной шиной Ethernet.

Следите за выходом новых продуктов, возможно, скоро появится модель с 2-гигабитным портом Ethernet.

Команда `pinout` устанавливается в составе пакета *python3-gpiozero*. Она включена в установку по умолчанию Raspberry Pi OS, но отсутствует в Raspberry Pi OS Lite. Установить ее можно с помощью команды `apt install python3-gpiozero`.

Дополнительная информация

- Схема контактов GPIO (https://oreil.ly/0pgXZ).
- Техническая информация и схемы для контроллера ENC28J60 (https://oreil.ly/HjlAM).

18.9. Настройка брандмауэра для общего использования подключения к Интернету с помощью firewalld

Задача

Настроить простой брандмауэр на Raspberry Pi, позволяющий совместно использовать подключение к Интернету и не пропускающий вредоносный трафик.

Решение

Используйте firewalld для фильтрации входящих пакетов, пропуская только ответы на трафик, исходящий из локальной сети, и запрещая запросы на подключение извне.

Настройка интернет-шлюза выглядит примерно так, как показано на рис. 18.7.



Рис. 18.7. Брандмауэр/маршрутизатор на Raspberry Pi

Большой и опасный Интернет приходит к вам через устройство, соединяющее вас с вашим интернет-провайдером. Подключите его к вашему Raspberry Pi, который будет фильтровать и направлять трафик в вашу локальную сеть через коммутатор Ethernet.

Для этого вам понадобится RPi с двумя сетевыми интерфейсами, один из которых следует подключить к устройству, соединяющему вас с Интернетом, а второй — к вашей локальной сети. В этом рецепте мы используем два интерфейса Ethernet.

Установите firewalld, и если будет такое желание, то еще и firewall-config и firewall-applet. firewall-config — это инструмент с графическим интерфейсом для настройки брандмауэра, a firewall-applet выводится на панели задач

и обеспечивает быстрый доступ к некоторым командам, таким как аварийная кнопка и кнопка блокировки:

```
$ sudo apt install firewalld firewall-config firewall-applet
```

Найдите свой маршрутизатор/шлюз по умолчанию:

```
$ ip r show
default via 192.168.1.1 dev eth0 proto dhcp src 192.168.1.43 metric 303 mtu 1500
192.168.1.0/24 dev eth0 proto dhcp scope link src 192.168.1.43 metric 303 mtu
1500cat
```

`default via 192.168.1.1` — это ваш шлюз по умолчанию.

Интерфейс `eth1` будет внешним интерфейсом, поэтому подключите его к своему устройству, соединяющему с Интернетом, а `eth0` будет внутренним интерфейсом — подключите его к коммутатору локальной сети. Интерфейсы должны находиться в разных подсетях. Интерфейс `eth1` должен находиться в одной подсети с устройством соединения с Интернетом. Допустим, что интерфейс локальной сети вашего устройства соединения с Интернетом имеет IP-адрес 192.168.1.1, тогда интерфейсу `eth1` можно присвоить адрес 192.168.1.2.

Интерфейс `eth0` должен находиться в адресном пространстве вашей локальной сети и иметь адрес, например, 192.168.2.1.

Настройте интерфейсы в файле `/etc/dhcpcd.conf`:

```
# внешний интерфейс
interface eth1
static ip_address=192.168.1.2/24
static routers=192.168.1.1

# внутренний интерфейс
interface eth0
static ip_address=192.168.2.1/24
static routers=192.168.1.1
```

Перезагрузите RPi, чтобы применить изменения.

Следующий шаг — настройка firewalld. Два интерфейса должны находиться в двух разных зонах брандмауэра: `eth1` должен находиться во внешней зоне (external), а `eth0` — во внутренней (intenal). После внесения изменений проверьте их:

```
$ sudo firewall-cmd --zone=external --change-interface=eth1
success
pi@raspberrypi:~ $ sudo firewall-cmd --zone=internal --change-interface=eth0
success
```

```
pi@raspberrypi:~ $ sudo firewall-cmd --get-active-zones
external
  interfaces: eth1
internal
  interfaces: eth0
```

Выведите настройки каждой зоны:

```
$ sudo firewall-cmd --zone=external --list-all
external (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth1
  sources:
  services: ssh
  ports:
  protocols:
  masquerade: yes
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:

$ sudo firewall-cmd --zone=internal --list-all
internal (active)
  target: default
  icmp-block-inversion: no
  interfaces: eth0 wlan0
  sources:
  services: dhcpv6-client mdns samba-client ssh
  ports:
  protocols:
  masquerade: no
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

Обратите внимание, что для внешней зоны по умолчанию открыт доступ по ssh. Вы можете добавлять или удалять любые службы, какие захотите. Оставьте включенным маскарадинг, поскольку именно он обеспечивает совместный доступ в Интернет.

Сделайте изменения постоянными:

```
$ sudo firewall-cmd --runtime-to-permanent
success
```

Пересылка (forwarding) IPv4 также включена по умолчанию во внешней зоне, в чем можно убедиться, заглянув в `/proc`. Пересылка IPv4 обеспечивает

маршрутизацию; в противном случае все пакеты, входящие в ваш RPi, не будут передаваться другим хостам в вашей сети.

```
$ cat /proc/sys/net/ipv4/ip_forward
1
```

`1` означает «включено», `0` — «выключено».

Комментарий

Маскарадингом IPv4 называется трансляция сетевых адресов или NAT (Network Address Translation). Трансляция сетевых адресов была создана для расширения ограниченного пула адресов IPv4 (который уже официально исчерпан). NAT позволяет свободно использовать частные адресные пространства IPv4 во внутренних сетях, обходясь без необходимости покупать общедоступные адреса IPv4. Ваш интернет-провайдер предоставляет вам хотя бы один общедоступный IPv4-адрес. Выполняя маскарадинг, брандмауэр преобразует частные адреса в единственный общедоступный IPv4-адрес; в противном случае хосты внутри локальной сети не будут иметь доступа к Интернету.

О добавлении и удалении служб из зон рассказывается в главе 14.

Дополнительная информация

- Глава 14.
- Отчет об ошибке #914694 в Debian (https://oreil.ly/SuHLL).

18.10. Запуск Raspberry Pi без монитора

Задача

Вы используете Raspberry Pi в качестве брандмауэра/маршрутизатора для подключения к Интернету, маршрутизатора в локальной сети какого-то легковесного сервера для локальной сети и хотите уменьшить нагрузку на него, запустив без графической среды рабочего стола.

Решение

Выполните следующие шаги.

1. Настройте доступ к RPi по SSH (см. главу 12).
2. Запустите `raspi-config` и запретите загрузку графического рабочего стола.

3. Перезагрузитесь.
4. Запустите команду `sudo raspi-config` и выберите в меню пункт 1 System Options ▸ S5 Boot/Auto Login (рис. 18.8).
5. Выберите загрузку в консоль без графического окружения, затем выполните перезагрузку.

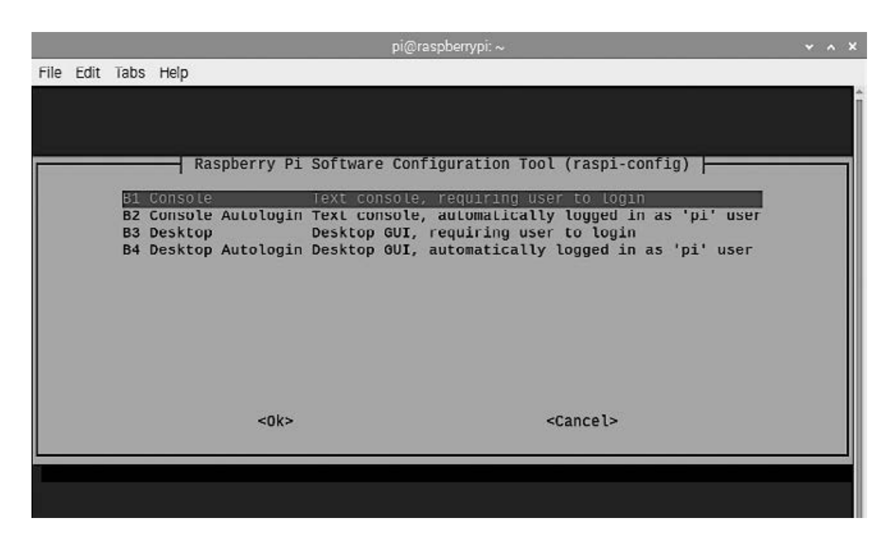


Рис. 18.8. Выберите загрузку в консоль без графической среды

Имея возможность начать сеанс SSH со своим RPi, вы сможете получить к нему доступ без монитора.

Если понадобится, то вы сможете запустить графическую среду из текстовой консоли, выполнив команду **`startx`**.

Комментарий

`rpi-config` использует `ncurses` — это консольный текстовый интерфейс, который выглядит как простой графический интерфейс.

Когда Raspberry Pi используется без монитора, ему нужны только питание и подключение к сети, а управлять им можно, установив сеанс SSH с другого компьютера.

Дополнительная информация

- Глава 12.
- The Raspberry Pi Foundation (https://oreil.ly/Ji4j2).

18.11. Создание сервера DNS/DHCP на Raspberry Pi

Задача

Ваше устройство соединения с Интернетом не предлагает возможности администрирования, и поэтому вы решили организовать свои локальные службы DNS и DHCP.

Решение

Отключите службы имен в своем устройстве соединения с Интернетом и настройте на втором Raspberry Pi службу имен Dnsmasq для локальной сети (см. главу 16). Используйте DHCP для распространения настроек всех служб и адресов между вашими хостами в локальной сети, включая статические адреса, за исключением интернет-шлюза, который не должен зависеть ни от каких внутренних служб.

Комментарий

Свой сервер имен можно также установить на свой брандмауэр/шлюз, но вообще размещение внутренних служб на хосте, напрямую подключенном к Интернету, считается плохим вариантом с точки зрения безопасности. Серверу DNS/DHCP на отдельном Raspberry Pi потребуется только один сетевой интерфейс, как и любому другому серверу локальной сети.

Дополнительная информация

- Глава 16.

ГЛАВА 19

Восстановление работоспособности системы с помощью SystemRescue

DVD или USB-накопитель SystemRescue — важный и нужный инструмент. Его можно использовать для восстановления не загружающихся систем Linux и Windows. В этой главе вы узнаете, как создать загрузочный носитель SystemRescue, ориентироваться в SystemRescue, настроить параметры загрузки, восстановить загрузчик GRUB, а также как извлечь файлы с неисправного диска, сбросить пароли в Linux и Windows и преобразовать файловую систему SystemRescue, доступную только для чтения, в файловую систему для чтения и записи с разделом для данных.

Любая live-версия Linux может использоваться для восстановления работоспособности системы. Преимуществом SystemRescue является его небольшой размер и подготовленность для выполнения восстановительных операций.

Корневая файловая система SystemRescue доступна только для чтения, поэтому любые внесенные вами изменения будут потеряны после перезагрузки. В данной главе вы узнаете, как настроить носитель с SystemRescue для сохранения ваших изменений, таких как настройки или дополнительное программное обеспечение.

SystemRescue изначально был основан на Gentoo Linux, а начиная с версии 6.0 — на Arch Linux. Arch Linux известен как надежный и эффективный дистрибутив, имеющий превосходную документацию, которую можно найти на сайте https://archlinux.org.

Я сама предпочитаю использовать USB-устройства для SystemRescue — флеш-накопители и жесткие диски. Они быстрые и имеют достаточно возможностей для копирования файлов.

19.1. Создание загрузочного устройства SystemRescue

Задача

Создать загрузочный DVD или USB-накопитель SystemRescue.

Решение

Скачайте последнюю версию `SystemRescue.iso` с сайта https://system-rescue.org.

Самый простой и надежный способ создать загрузочную флешку SystemRescue — использовать команду `dd` (см. рецепт 1.6). Инструкции по созданию загрузочного DVD вы найдете в рецептах 1.4 и 1.5. В главе 1 также рассказывается как загрузиться с нового носителя и как отключить безопасную загрузку. SystemRescue не имеет подписанных ключей, поэтому необходимо отключить безопасную загрузку в UEFI.

Чтобы загрузиться с USB-накопителя, подключите его к USB-порту на своем компьютере, а не к USB-концентратору, поскольку при подключении к последнему носитель не будет распознан.

Комментарий

Не забудьте снова включить безопасную загрузку, когда закончите использовать SystemRescue.

Дополнительная информация

- https://system-rescue.org.
- Глава 1.

19.2. Начало работы с SystemRescue

Задача

Вы загрузились с носителя SystemRescue, оказались в простой командной строке и теперь хотите знать, что делать дальше.

Решение

SystemRescue выводит инструкции на начальном экране входа в систему (рис. 19.1). Вы автоматически входите в систему как root, причем пароль root отсутствует.

```
========= SystemRescue 8.00 (x86_64) ======== tty1/6 =========
                  https://www.system-rescue.org/

 * Console environment :
   Run setkmap to choose the keyboard layout

 * Graphical environment :
   Type startx to run the graphical environment
   X.Org comes with the XFCE environment and several graphical tools:
   - Partition manager: .. gparted
   - Web browser: ........ firefox
   - Text editor: ........ featherpad

sysrescue login: root (automatic login)

[root@sysrescue ~]# _
```

Рис. 19.1. Экран входа в SystemRescue

Вы можете продолжать работать в консольном режиме или ввести команду **`startx`**, чтобы запустить среду рабочего стола Xfce4 (рис. 19.2).

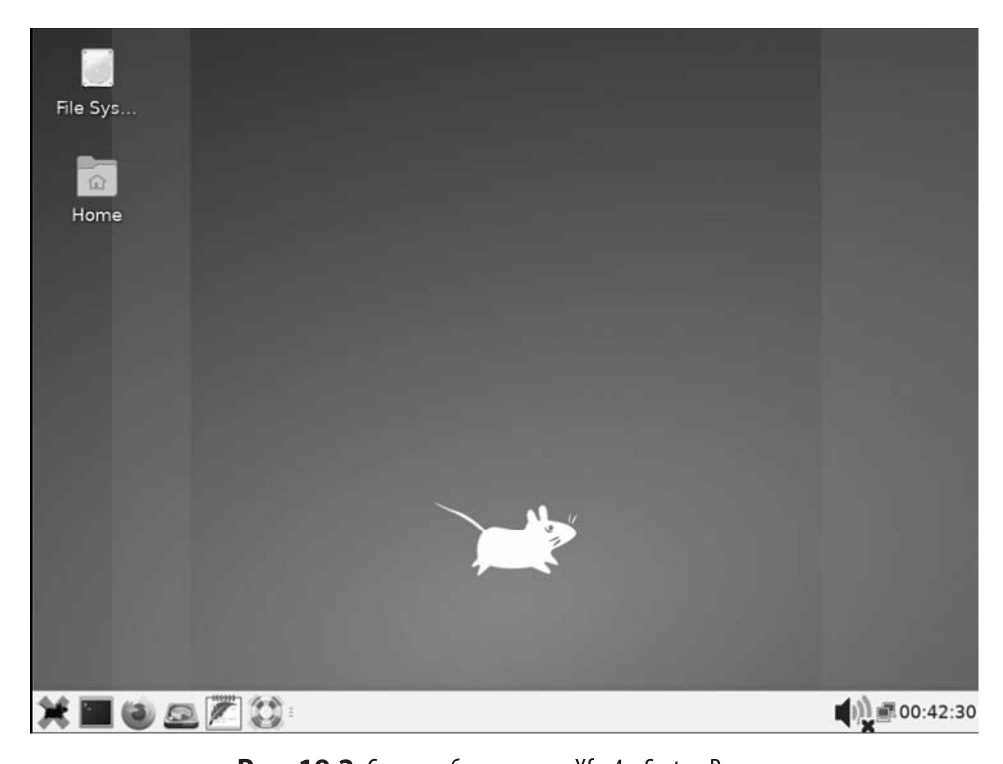


Рис. 19.2. Среда рабочего стола Xfce4 в SystemRescue

В SystemRescue можно исследовать меню приложений, настроить внешний вид Xfce, подключиться к сети с помощью NetworkManager, остановить или перезагрузить систему, как в любой другой системе Linux.

Комментарий

Единственное, чего нельзя сделать при использовании стандартного образа SystemRescue, — это вносить постоянные изменения в его конфигурацию. Любые внесенные изменения исчезнут после перезапуска, поскольку SystemRescue запускается из сжатой файловой системы SquashFS, доступной только для чтения. Однако есть возможность настроить образ так, чтобы изменения сохранялись (см. рецепт 19.14).

SquashFS — основа многих live-версий Linux, таких как Ubuntu, Debian, Mint, Fedora и Arch. Она также используется проектами, создающими прошивки DDWRT и OpenWRT с открытым исходным кодом для маршрутизаторов. SquashFS — легковесная и быстрая файловая система.

Мне нравится Xfce4 — легковесная графическая среда рабочего стола, поддерживающая приложения с графическим интерфейсом и включающая X-терминал для выполнения операций из командной строки.

Дополнительная информация

- https://system-rescue.org.
- https://xfce.org.

19.3. Знакомство с двумя загрузочными меню SystemRescue

Задача

Во время опробования вы заметили, что в SystemRescue есть два разных меню с вариантами загрузки, и теперь хотите понять, для чего они нужны.

Решение

В зависимости от способа загрузки SystemRescue отображает два разных меню с вариантами загрузки. Одно (рис. 19.3) отображается, когда загрузка производится устаревшим способом BIOS, а другое (рис. 19.4) — при загрузке UEFI.

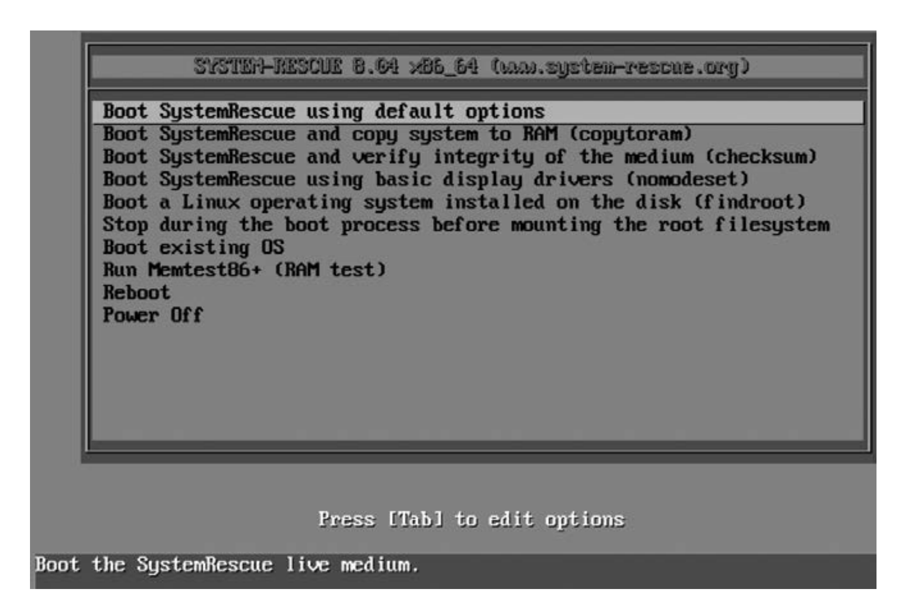


Рис. 19.3. Загрузочное меню SystemRescue при использовании устаревшего способа загрузки BIOS

```
                    GNU GRUB  version 2.06

*Boot SystemRescue using default options
 Boot SystemRescue and copy system to RAM (copytoram)
 Boot SystemRescue and verify integrity of the medium (checksum)
 Boot SystemRescue using basic display drivers (nomodeset)
 Boot a Linux operating system installed on the disk (findroot)
 Stop during the boot process before mounting the root filesystem
 Start EFI Shell
 EFI Firmware setup
 Reboot
 Power off

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, `e' to edit the commands
   before booting or `c' for a command-line.
```

Рис. 19.4. Загрузочное меню SystemRescue при использовании способа загрузки UEFI

Например, когда в настройках UEFI моего компьютера Dell разрешена загрузка с устаревших устройств, то в меню загрузки (отображается при нажатии клавиши F12 при запуске) выводятся все доступные варианты загрузки (рис. 19.5). Однако SystemRescue поддерживает UEFI, поэтому включать загрузку с устаревших устройств не обязательно. (Но не забывайте, что для загрузки SystemRescue необходимо отключить безопасную загрузку.)

```
Use the ↑(Up) and ↓(Down) arrow keys to move the pointer to the desired boot device.
Press [Enter] to attempt the boot or ESC to Cancel. (* = Password Required)

Boot mode is set to: UEFI; Secure Boot: OFF

LEGACY BOOT:
    SAMSUNG SSD SM871 2.5 7mm
    ST1000DM003-1SB102
    P0: PLDS DVD+/-RW DH-16AES
UEFI BOOT:
    opensuse-secureboot
    Windows Boot Manager
    UEFI: ST1000DM003-1SB102
OTHER OPTIONS:
    BIOS Setup
    BIOS Flash Update
    Diagnostics
    Intel(R) Management Engine BIOS Extension (MEBx)
    Change Boot Mode Settings
```

Рис. 19.5. Все возможные способы загрузки в загрузочном меню на компьютере Dell

Настройки BIOS/UEFI в вашей системе могут отличаться от моих, поскольку все компьютеры разные.

Главное отличие этих двух меню друг от друга — наличие в загрузочном меню UEFI пунктов Start EFI Shell (Запустить оболочку EFI) и EFI Firmware Setup (Настройка прошивки EFI), которые неприменимы к BIOS.

В загрузочном меню BIOS нет пунктов, имеющих отношение к EFI, зато есть пункт Memtest86+, запускающий проверку системной памяти.

Подробнее о разных вариантах загрузки SystemRescue рассказывается в рецепте 19.4.

Комментарий

При желании можно настроить загрузку по умолчанию со съемного носителя. Тогда вам не придется помнить о необходимости входа в BIOS/UEFI, чтобы выбрать другое загрузочное устройство, или ждать подходящего момента для вызова загрузочного меню; просто вставьте съемный носитель, когда это понадобится, и загрузитесь с него.

Если вы используете старую систему без UEFI, то вам не придется выбирать или возиться с безопасной загрузкой.

Дополнительная информация

- https://system-rescue.org.
- Глава 1.
- Рецепт 19.4.

19.4. Знакомство с вариантами загрузки SystemRescue

Задача

Узнать назначение всех пунктов в загрузочном меню SystemRescue (см. рецепт 19.3).

Решение

Пункты в загрузочном меню соответствуют наиболее часто используемым вариантам загрузки, избавляющим от необходимости открывать форму редактирования и вводить их. Чаще других используется первый пункт: Boot SystemRescue Using Default Options (Загрузить SystemRescue с параметрами по умолчанию).

Второй пункт — Boot SystemRescue and copy to RAM (copytoram) (Загрузить и скопировать SystemRescue в ОЗУ (copytoram)) — улучшает производительность за счет полной загрузки образа SystemRescue в память. Выигрыш особенно заметен при запуске SystemRescue с DVD. Для образа используется около 2 Гбайт ОЗУ.

Третий пункт — Boot SystemRescue and verify integrity of the medium (checksum) (Загрузить SystemRescue и проверить целостность носителя (checksum)) — позволяет проверить носитель с образом на наличие повреждений. Используйте его, чтобы убедиться, что ваш носитель SystemRescue исправен.

Четвертый пункт — Boot SystemRescue using basic display drivers (nomodeset) (Загрузить SystemRescue с базовыми видеодрайверами (nomodeset)) — использует базовые видеодрайверы, поддерживающие более низкие разрешения. Выбирайте этот пункт, если изображение на экране выглядит некорректно из-за отсутствия в SystemRescue нужных драйверов.

Пятый пункт — Boot a Linux operating system installed on the disk (findroot) (Загрузить операционную систему Linux, установленную на диске (findroot)) — хороший способ проверить, не является ли загрузчик причиной невозможности загрузить Linux. При выборе этого пункта SystemRescue найдет загрузочный раздел, и если их несколько, то перечислит их все, а вы сможете выбрать, какой использовать.

Шестой пункт — Stop during the root process before mounting the root filesystem (Остановить процесс загрузки перед монтированием корневой файловой системы) — это режим восстановления на случай, если SystemRescue не загружается. Как мне кажется, проще иметь под рукой несколько дополнительных дисков с SystemRescue, чем пытаться восстановить неисправный SystemRescue.

В загрузочном меню UEFI есть два дополнительных пункта: Start EFI Shell (Запустить оболочку EFI) и EFI Firmware Setup (Настройка прошивки EFI). Первый открывает доступ к многочисленным утилитам EFI, а второй — к настройке UEFI вашей системы.

Последними следуют пункты Reboot (Перезагрузить) и Power off (Выключить питание).

В загрузочном меню BIOS есть два дополнительных пункта: Boot existing OS (Загрузить существующую ОС) и Run Memtest86+ (Запустить Memtest86+). Первый помогает диагностировать проблемы с загрузчиком, позволяя загрузить систему в обход штатного загрузчика, а второй запускает проверку системной памяти.

Оба меню включают пункты Reboot (Перезагрузить) и Power off (Выключить питание) для перезагрузки или выключения питания вместо загрузки SystemRescue.

Комментарий

Я не вижу особого смысла в использовании оболочки EFI, поскольку она поддерживает расширенные операции, не требующиеся для восстановления незагружающейся системы. См. руководство компании Intel по использованию расширяемого микропрограммного интерфейса Basic Instructions for Using the Extensible Firmware Interface (https://oreil.ly/dktzy), чтобы получить полную информацию.

Все эти пункты меню соответствуют наиболее часто используемым параметрам загрузки SystemRescue, которые описаны на https://system-rescue.org. Вы можете добавить параметры загрузки в любой из пунктов загрузочного меню SystemRescue, как будет показано в нескольких рецептах в данной главе.

Все эти задачи можно выполнить после запуска SystemRescue или организовать их выполнение на этапе загрузки с помощью параметров. На загрузки с BIOS выберите пункт меню и затем нажмите клавишу TAB, после чего откроется поле редактирования. Введите **`rootpass=`*`ваш_пароль`*` nofirewall`**, затем нажмите Enter, чтобы продолжить запуск.

Чтобы определить свои параметры в меню загрузки для UEFI, нажмите клавишу E.

Дополнительная информация

- https://system-rescue.org.

19.5. Идентификация файловых систем

Задача

Узнать, как идентифицировать имеющиеся файловые системы на жестких дисках, чтобы правильно выбирать нужные для проведения восстановительных мероприятий.

Решение

Используйте старую добрую команду `lsblk`:

```
# lsblk -f
NAME   FSTYPE   FSVER LABEL     UUID                 SAVAIL FSUSE% MOUNTPOINT
loop0  squashfs 4.0                                       0   100% /run/archiso/sf
s/airootfs
sda
├─sda1
└─sda2 ntfs                     5E363
sdb
├─sdb1 vfat     FAT16 BOOT      5E2F-1E75
├─sdb2 btrfs          root      02bfdc9a-b8bb-45ac-95a8
├─sdb3 xfs            home      cc8acf0b-529e-473c-b484
└─sdb4 swap     1               7a5519ae-efe6-45e6-b147
sdc    iso9660        RESCUE800 2021-03-06-08-53-50-00
└─sdc1 iso9660        RESCUE800 2021-03-06-08-53-50-00    0   100% /run/archiso/
bootmnt
```

Комментарий

Использование меток файловых систем значительно упрощает их поиск. Кроме того, метки можно применять вместо UUID, например, в файле `/etc/fstab`. Более подробно об управлении метками файловых систем рассказывается в рецепте 9.4 и в главе 11.

Команда `lsblk` не требует привилегий суперпользователя и отображает почти всю информацию о блочных устройствах, которая может понадобиться.

Дополнительная информация

- Рецепт 9.4.
- Рецепт 11.2.
- Глава 11.

19.6. Переустановка пароля root в Linux

Задача

Вы забыли свой пароль root и хотите переустановить его.

Решение

Загрузите SystemRescue и смонтируйте корректную корневую файловую систему. В следующих примерах корневая файловая система находится в разделе `/dev/sdb2`. Создайте точку монтирования в `/mnt`, затем смонтируйте файловую систему:

```
# mkdir /mnt/sdb2
# mount /dev/sdb2 /mnt/sdb2
```

Сделайте смонтированную файловую систему корневой для SystemRescue:

```
# chroot /mnt/sdb2/ /bin/bash
:/ #
```

Сбросьте пароль root:

```
:/ # passwd root
New password:
Retype new password:
passwd: password updated successfully
:/ #
```

Введите команду `exit`, чтобы вернуть на место корневую файловую систему SystemRescue.

Перезагрузитесь и попробуйте войти с новым паролем.

Комментарий

Забытый пароль нельзя восстановить, можно только создать новый.

Этот прием позволяет переустанавливать пароль любого пользователя.

Сменив корневую файловую систему, вы сможете выполнять некоторые команды, но не все, поскольку это не совсем полноценная корневая файловая система — в ней отсутствуют все псевдофайловые системы, существующие только в памяти, такие как *sysfs* и *proc*. См. рецепт 19.9, чтобы узнать, как настроить более полноценную среду *chroot*.

Когда-то давно можно было сбросить пароль root, удалив его хеш в `/etc/shadow`. Однако современная подсистема *pam*, управляющая авторизацией,

намного сложнее. Если вам будет интересно, то изучите конфигурацию pam в SystemRescue, чтобы узнать, как она настроена на использование пустого пароля root.

Дополнительная информация

- https://system-rescue.org.
- Глава 5.
- `man 7 pam`

19.7. Включение поддержки SSH в SystemRescue

Задача

Получить доступ к SystemRescue по SSH.

Решение

Поддержка SSH включена по умолчанию, как и брандмауэр. Отключите его, чтобы разрешить входящие SSH-соединения.

После загрузки SystemRescue отключите брандмауэр с помощью команды `systemctl`:

```
[root@systemrescue ~]# systemctl stop iptables.service
```

По умолчанию SystemRescue не имеет пароля для пользователя root, поэтому вы должны создать его, чтобы обеспечить возможность запустить сеанс SSH:

```
[root@systemrescue ~]# passwd root
New password:
Retype new password:
passwd: password updated successfully
```

Теперь можно выполнить вход в SystemRescue с другого компьютера:

```
$ ssh root@192.168.10.101
ssh root@192.168.1.91
The authenticity of host '192.168.1.91 (192.168.1.91)' can't be established.
ECDSA key fingerprint is SHA256:LlUCEngz5NHg98xv.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.91' (ECDSA) to the list of known hosts.
root@192.168.1.91's password:
[root@sysrescue ~]#
```

Комментарий

Каждый раз, загружая SystemRescue, вы получаете абсолютно новую систему с уникальным ключом SSH. Перезагрузив SystemRescue и открыв второй сеанс SSH с того же компьютера, вы увидите следующее предупреждение:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
```

Далее следуют еще несколько строк, и в конце рекомендуется возможное исправление проблемы:

```
Offending ECDSA key in /home/duchess/.ssh/known_hosts:12
  remove with:
  ssh-keygen -f "/home/duchess/.ssh/known_hosts" -R "192.168.10.101"
```

Последуйте совету и тогда сможете подключиться к SystemRescue по SSH.

Отключить брандмауэр можно из меню загрузки. Нажмите клавишу Tab (загрузка из BIOS) или клавишу E (загрузка из UEFI), чтобы добавить параметр `nofirewall` (см. строку параметров загрузки, которая показана в нижней части экрана на рис. 19.6).

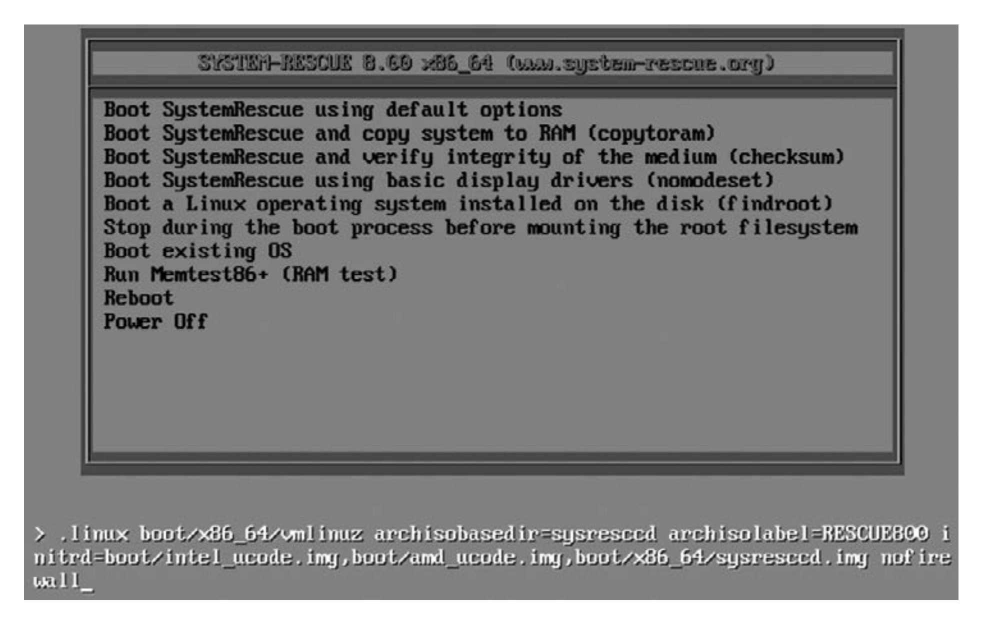


Рис. 19.6. Отключение брандмауэра перед загрузкой

Дополнительная информация

- https://system-rescue.org.
- Рецепт 19.4.
- Глава 12.

19.8. Копирование файлов по сети с помощью scp и sshfs

Задача

Вы загрузились с носителя SystemRescue и хотите скопировать некоторые файлы из аварийной системы по сети.

Решение

Это легко сделать, как в любой другой системе Linux. Сначала включите SSH (см. рецепт 19.7). Затем используйте команду `scp` или утилиту sshfs для копирования выбранных файлов. Все команды в этом рецепте запускаются из SystemRescue.

С помощью команды `lsblk` найдите файловую систему, в которой располагаются файлы для копирования. Если вы не помните, какой раздел вам нужен, то монтируйте каждый из них по очереди и просматривайте содержимое, пока не найдете нужный:

```
# lsblk -f
NAME   FSTYPE   FSVER LABEL     UUID                 SAVAIL FSUSE% MOUNTPOINT
loop0  squashfs 4.0                                       0   100% /run/archiso/
sf
s/airootfs
sda
├─sda1
└─sda2 ntfs                     5E363E30363E0993
sdb
├─sdb1 vfat     FAT16 BOOT      5E2F-1E75
├─sdb2 btrfs          root      02bfdc9a-b8bb-45ac-95a8
├─sdb3 xfs            home      cc8acf0b-529e-473c-b484
└─sdb4 swap     1               7a5519ae-efe6-45e6-b147
sdc    iso9660        RESCUE800 2021-03-06-08-53-50-00
└─sdc1 iso9660        RESCUE800 2021-03-06-08-53-50-00    0   100% /run/
archiso/b
ootmnt
sr0
```

В следующем примере показано, как смонтировать раздел с каталогом /home в аварийной системе в каталог /mnt в SystemRescue, вывести список файлов и затем скопировать весь каталог /home на компьютер пользователя Duchess с помощью команды scp:

```
# mkdir /mnt/sdb3
# mount /dev/sdb3 /mnt/sda3
# ls /mnt/sdb3
bin   dev  home  lib64       media  opt   root  sbin  sys  usr
boot  etc  lib   lost+found  mnt    proc  run   srv   tmp  var
# scp -r /mnt/sdb3/home/ duchess@pc:
```

Скопированный каталог будет сохранен в /home/duchess/home.



Всегда создавайте подкаталог в /mnt

Никогда не монтируйте файловые системы непосредственно в каталог /mnt; это сделает невозможной нормальную работу SystemRescue. Всегда создавайте подкаталоги для точек монтирования.

При желании можно указать список копируемых файлов, перечислив их через пробел. Кроме того, в одном списке можно смешивать имена файлов и каталогов. В следующем примере копирование выполняется в каталог rescue на компьютер duchess@pc. Удаленный каталог должен существовать:

```
# cd /mnt/sdb3/home/
# scp -r file1.txt directory1 file2.txt duchess@pc:rescue/
```

Утилита sshfs удобна тем, что монтирует удаленную файловую систему, благодаря чему та выглядит как локальная ФС, и вы можете копировать в нее файлы так же, как в любую другую локальную систему. Создайте точку монтирования в SystemRescue, затем смонтируйте в нее удаленный каталог, куда предполагается скопировать файлы. Удаленный каталог уже должен существовать:

```
# mkdir /mnt/remote
# sshfs duchess@pc:rescue/ /mnt/remote/
# ls /mnt/remote
rescue
```

После этого можно использовать команду cp в SystemRescue или графический диспетчер файлов (рис. 19.7).

По завершении выполните **fusermount -u *remote***, чтобы размонтировать файловую систему sshfs.

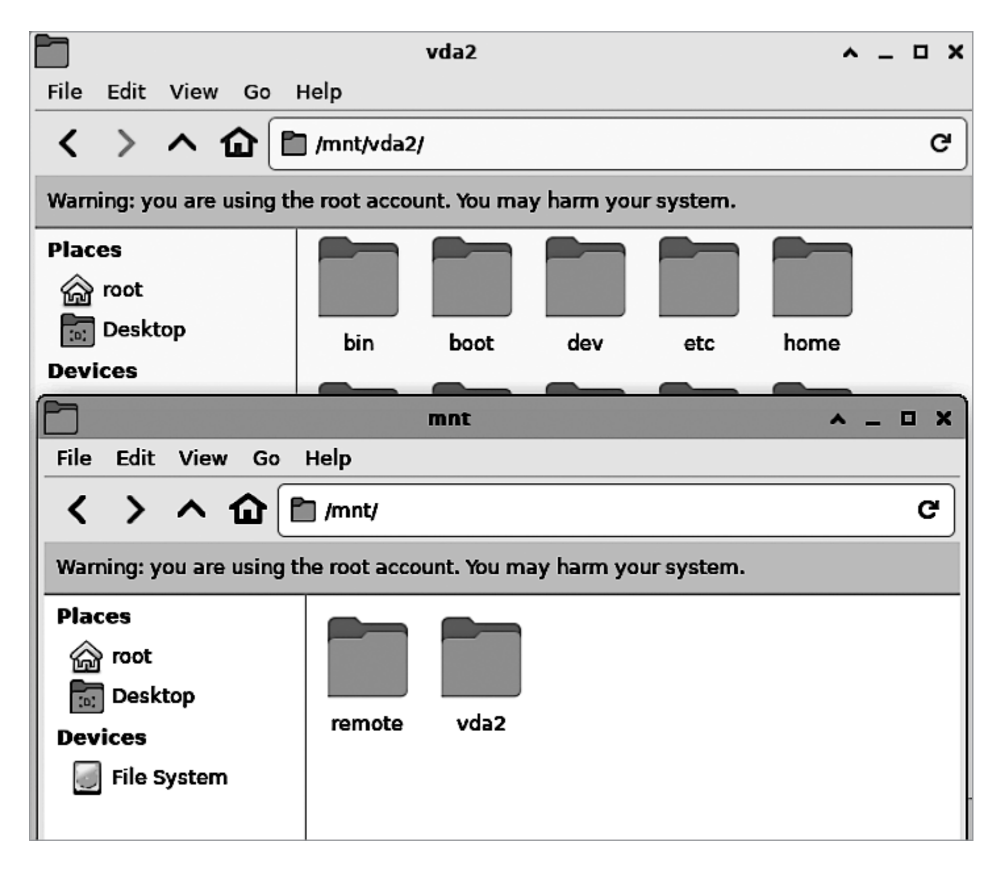


Рис. 19.7. Копирование файлов с помощью графического диспетчера файлов в SystemRescue

Комментарий

Если вы отключили аутентификацию по паролю SSH в системе, куда собираетесь копировать файлы, то включите ее на время, закомментировав `PermitRootLogin no` в файле `/etc/ssh/sshd_config`.

Синтаксис подключения к удаленному каталогу зависит от учетной записи пользователя, в которую вы входите. `duchess@pc:` — это то же самое, что и `duchess@pc:/home/duchess`. Путь `duchess@pc:/` соответствует корневой файловой системе. Если вам нужно отредактировать конфигурационные файлы системы, то используйте путь `duchess@pc:/etc`; для доступа к файлам загрузчика используйте `duchess@pc:/boot` и т. д.

Создать удаленный каталог из SystemRescue можно с помощью `ssh`:

```
# ssh duchess@pc
duchess@pc's password:
duchess@pc:~$ mkdir remote
```

Дополнительная информация

- Рецепт 6.5.
- Глава 12.

19.9. Восстановление загрузчика GRUB из SystemRescue

Задача

Работоспособность загрузчика GRUB была нарушена, и система перестала загружаться.

Решение

Загрузитесь с носителя SystemRescue, создайте среду chroot и переустановите GRUB.

После загрузки SystemRescue создайте среду chroot для корневой файловой системы хоста:

```
# mkdir /mnt/linux
# mount /dev/sda2 /mnt/linux
# mount -o bind /proc /mnt/linux/proc
# mount -o bind /dev /mnt/linux/dev
# mount -o bind /sys /mnt/linux/dev
```

Войдите в среду chroot:

```
# chroot /mnt/linux /bin/bash
:/ #
```

Если каталог `/boot` размещен в отдельном разделе, то смонтируйте его:

```
:/ # mount /dev/sda1 /boot/
```

Затем переустановите GRUB:

```
:/ # grub-install /dev/sda
```

По завершении введите команду `exit`, чтобы выйти из среды chroot, затем размонтируйте все файловые системы и перезагрузите систему. GRUB должен работать.

Комментарий

Будьте очень осторожны при создании среды chroot и убедитесь, что используете правильные разделы и файловые системы. chroot — сокращенно от *change root* (сменить корень) — отличная утилита для перехода на другую корневую файловую систему без перезагрузки.

Вы должны размонтировать все файловые системы chroot, чтобы потом не возникло никаких проблем. Конечно, можно просто перезагрузить компьютер, но предварительное размонтирование вручную дает дополнительную страховку от неожиданностей.

Дополнительная информация

- `man 1 chroot`

19.10. Переустановка пароля в Windows

Задача

Вы забыли пароль к Windows и не хотите идти стандартным путем его переустановки.

Решение

Не беспокойтесь, SystemRescue поможет вам сделать это в мгновение ока. Загрузитесь с носителя SystemRescue на компьютере с Windows, затем смонтируйте системный каталог Windows:

```
# mkdir /mnt/windows
# mount /dev/sda2
```

Перейдите в каталог `/mnt/windows/Windows/System32/config`, затем введите команду `chntpw` (change NT password — «изменить пароль NT»), чтобы вывести список пользователей:

```
# cd /mnt/windows/Windows/System32/config
# chntpw -l SAM
chntpw version 1.00 140201, (c) Petter N Hagen
Hive <SAM> name (from header): <\SystemRoot\System32\Config\SAM>
ROOT KEY at offset: 0x001020 * Subkey indexing type is: 686c <lh>
File size 65536 [10000] bytes, containing 7 pages (+ 1 headerpage)
Used for data: 318/31864 blocks/bytes, unused: 29/12968 blocks/bytes.

| RID -|---------- Username ------------| Admin? |- Lock? --|
| 01f4 | Administrator                  | ADMIN  |          |
| 03e9 | duchess                        | ADMIN  |          |
| 01f7 | DefaultAccount                 |        | dis/lock |
| 01f5 | Guest                          |        | dis/lock |
| 01f8 | WDAGUtilityAccount             |        | dis/lock |
```

Исследуйте информацию о пользователе, пароль которого вы хотите изменить:

```
# chntpw -u Administrator SAM
chntpw version 1.00 140201, (c) Petter N Hagen
Hive <SAM> name (from header): <\SystemRoot\System32\Config\SAM>
ROOT KEY at offset: 0x001020 * Subkey indexing type is: 686c <lh>
File size 65536 [10000] bytes, containing 9 pages (+ 1 headerpage)
Used for data: 321/33816 blocks/bytes, unused: 34/27336 blocks/bytes.

================= USER EDIT ====================

RID     : 0500 [01f4]
Username: Administrator
fullname:
comment : Built-in account for administering the computer/domain
homedir :

00000220 = Administrators (which has 2 members)

Account bits: 0x0210 =
[ ] Disabled        | [ ] Homedir req.    | [ ] Passwd not req. |
[ ] Temp. duplicate | [X] Normal account  | [ ] NMS account     |
[ ] Domain trust ac | [ ] Wks trust act.  | [ ] Srv trust act   |
[X] Pwd don't expir | [ ] Auto lockout    | [ ] (unknown 0x08)  |
[ ] (unknown 0x10)  | [ ] (unknown 0x20)  | [ ] (unknown 0x40)  |

Failed login count: 0, while max tries is: 0
Total  login count: 5

- - - - User Edit Menu:
 1 - Clear (blank) user password
 2 - Unlock and enable user account [probably locked now]
```

```
 3 - Promote user (make user an administrator)
 4 - Add user to a group
 5 - Remove user from a group
 q - Quit editing user, back to user select
Select: [q] ^
```

Введите **1**, чтобы удалить текущий пароль:

```
Select: [q] ^ 1
Password cleared!
[...]
```

Нажмите **q**, чтобы выйти, и **y**, чтобы сохранить изменения.

Теперь администратор или другой выбранный вами пользователь должен войти в систему и установить новый пароль.

Комментарий

С помощью команды `chntpw` нельзя создать новый пароль или восстановить старый, можно только удалить его. После этого станет доступен вход в систему без пароля, и вы сможете создать новый пароль или, если ваш пользователь на месте, дать сделать это ему.

Дополнительная информация

- https://system-rescue.org.
- `man 8 chntpw`

19.11. Восстановление аварийного жесткого диска с помощью GNU ddrescue

Задача

Вы подозреваете, что ваш жесткий диск вот-вот выйдет из строя, и хотите побыстрее скопировать с него данные, которые еще можно спасти.

Решение

Вам понадобится замечательная утилита GNU *ddrescue*. Она сначала попытается скопировать все хорошие блоки и сохранить как можно больше данных,

пропуская плохие блоки и запоминая их расположение в файле журнала. Вы можете попробовать сделать несколько проходов, чтобы попытаться собрать больше данных.

Диск, данные с которого вы хотите скопировать, должен быть размонтирован. Вам понадобится другой диск, тоже размонтированный, например USB-накопитель или внутренний жесткий диск, на который можно скопировать восстановленные данные. Целевой раздел должен уже существовать и быть как минимум на 50 % больше сохраняемого раздела.

Следующий пример копирует `/dev/sdb1` в `/dev/sdc1`:

```
# ddrescue -f -n /dev/sdb1 /dev/sdc1 ddlogfile
GNU ddrescue 1.25
Press Ctrl-C to interrupt
     ipos:   100177 MB, non-trimmed: 0 B    current rate:   207 MB/s
     opos:   100177 MB, non-scraped: 0 B    average rate: 83686 kB/s
non-tried:    47868 MB,  bad-sector: 0 B,     error rate:      0 B/s
  rescued:   100177 MB,   bad areas: 0,         run time:    23m 56s
pct rescued:    66.77%, read errors: 0,   remaining time:      6m 4s
                         time since last successful read:         0s
Copying non-tried blocks... Pass 1 (forwards)
```

Это займет некоторое время. По завершении в последней строке появится текст Finished (Готово).

В данном примере выполняется один проход, чтобы максимально быстро скопировать наиболее легко читаемые блоки. Это хорошая тактика для диска с большим количеством ошибок, поскольку ddrescue не станет тратить много времени на восстановление особенно сильно поврежденных блоков. После первого прохода запустите еще три прохода, чтобы попытаться восстановить больше данных:

```
# ddrescue -d -f -r3 /dev/sdb1 /dev/sdc1 ddlogfile
```

По завершении запустите проверку файловой системы на диске, куда были скопированы данные, он при этом должен оставаться размонтированным. В следующем примере проверяется и автоматически восстанавливается файловая система Ext4:

```
# e2fsck -vfp /dev/sdc1
```

Параметр `-f` включает режим принудительной проверки на случай, если `e2fsck` сочтет файловую систему исправной. `-p` означает preen («восстано-

вить»), а `-v` включает режим подробного вывода. Если обнаружится проблема, требующая вашего вмешательства, то утилита выведет описание проблемы и завершит работу.

`e2fsck -vf` *[устройство]* запускает проверку и восстановление в интерактивном режиме.

`fsck.vfat -vfp` *[устройство]* — то же самое для FAT16/32.

`xfs_repair` *[устройство]* — для XFS.

Если копия файловой системы прошла проверки, то сделайте следующий шаг и скопируйте файлы в окончательное местоположение. При наличии же неких проблем смонтируйте ее в режиме только для чтения:

```
# mkdir /mnt/sdc1-copy
# mount -o ro /dev/sdc1 /mnt/sdc1-copy
```

И затем скопируйте файлы, какие сможете, на другой диск.

Комментарий

Убедитесь, что используете утилиту GNU ddrescue, которую написал Антонио Диас Диас (Antonio Diaz Diaz), а не ddrescue Курта Гарлоффа (Kurt Garloff) (это отличный инструмент, но более сложный в использовании).

Утилита ddrescue копирует данные на уровне блоков, поэтому тип файловой системы не имеет значения. Утилита создаст точную копию независимо от того, какие файловые системы поддерживает ваша Linux.

Если ddrescue не хватит места, то она завершится с ошибкой в самом конце, поэтому убедитесь, что на диске для сохранения восстановленной копии достаточно места.

Утилита ddrescue поддерживает USB-накопители, CompactFlash и SD-карты.

Дополнительная информация

- GNU ddrescue (https://oreil.ly/mMxQf).
- `man 8 fsck` (`e2fsprogs`)

19.12. Управление разделами и файловыми системами из SystemRescue

Задача

Разметить жесткий диск или внести изменения в файловую систему, причем сделать это нужно из внешней (то есть не из установленной на компьютере) системы Linux.

Решение

Используйте SystemRescue. В его состав входят не только `parted`, но и GParted. Вам не нужно монтировать какие-либо файловые системы, и SystemRescue будет работать непосредственно с блочными устройствами хоста. Запустите команду `lsblk`, чтобы увидеть список имеющихся блочных устройств хоста:

```
[root@systemrescue ~]# lsblk -p -o NAME,FSTYPE,LABEL
NAME           FSTYPE     LABEL
/dev/loop/0    squashfs
/dev/sr0
/dev/sr1       iso9660    RESCUE800
/dev/sda
├─/dev/sda1    vfat
├─/dev/sda2    xfs        osuse15-2
├─/dev/sda3    xfs        home
├─/dev/sda4    xfs
└─/dev/sda5    swap
/dev/sdb
└─/dev/sdb1    xfs        backups
/dev/sr0
```

Последуйте рецептам управления разделами и файловыми системами, которые приводятся в главах 8, 9 и 11.

Дополнительная информация

- Глава 8.
- Глава 9.
- Глава 11.

19.13. Создание раздела для данных на USB-носителе с SystemRescue

Задача

SystemRescue на USB-носителе — хороший инструмент, но хотелось бы разбить данный носитель на разделы и в первом разместить корневую файловую систему SystemRescue, а во втором создать файловую систему с поддержкой записи для хранения своих данных. В этом случае вам вполне может хватить одного устройства для копирования файлов.

Решение

Это можно сделать всего за несколько шагов.

Стандартный образ SystemRescue не может загрузиться из раздела. Но он занимает меньше гигабайта, поэтому на любом современном USB-носителе остается недоступным масса пустого пространства. Хитрость, позволяющая сделать возможной загрузку SystemRescue из раздела, состоит в том, чтобы создать ISO-образ с SystemRescue, который может загружаться из раздела, добавить основную загрузочную запись (master boot record, MBR), а затем установить загрузочный код *mbr.bin*.

Для этого вам понадобятся *isohybrid* и *mbr.bin*, которые входят в состав набора утилит *syslinux*. Этот набор распространяется в виде пакета *syslinux* в Fedora и openSUSE и *syslinux-utils* и *install-mbr* — в Ubuntu.

В следующих примерах замените `/dev/sdc` на имя своего устройства.

Сначала создайте образ SystemRescue, который может загружаться из раздела:

```
$ isohybrid --partok systemrescuecd-8.01-amd64.iso
```

Создайте таблицу разделов `msdos` на USB-носителе. Используйте для этого GParted (см. рецепт 9.2) или `parted`:

```
$ sudo parted /dev/sdc
(parted) mklabel msdos
```

Создайте два раздела на USB-носителе. Для первого раздела выберите тип файловой системы FAT32 и установите флаг `boot` (загрузочный). Следующий

пример демонстрирует создание загрузочного раздела размером примерно 2 Гбайт:

```
(parted) mkpart "sysrec" fat32 1MB 2000MB
(parted) set 1 boot
```

Создайте второй раздел для хранения данных. Для него можно выбрать любую файловую систему, которая вам по душе. Следующий пример демонстрирует создание раздела размером 2 Гбайт и завершает работу `parted`:

```
(parted) mkpart "data" xfs 2001MB 4000MB
(parted) q
```

Создайте файловые системы. Ниже представлен пример создания в разделе 1 файловой системы FAT32 с меткой `SYSRESCUE` и в разделе 2 — файловой системы XFS с меткой `data`:

```
$ sudo mkfs.fat -F 32 -n SYSRESCUE /dev/sdc1
$ sudo mkfs.xfs -L data /dev/sdc2
```

Установите SystemRescue в первый раздел:

```
$ sudo dd status=progress if=systemrescuecd-8.01-amd64.iso of=/dev/sdc1
```

В Ubuntu установите MBR на USB-носитель:

```
$ sudo install-mbr /dev/sdc
```

В других дистрибутивах используйте `dd`:

```
$ sudo if=/usr/share/syslinux/mbr.bin of=/dev/sdc
```

`mbr.bin` может находиться в другом каталоге, в зависимости от используемого вами дистрибутива Linux.

Загрузитесь с носителя SystemRescue, и система должна запуститься как обычно.

Комментарий

Метки файловых систем, показанные в этих примерах, создавать не обязательно, но они помогают вспомнить назначение данных файловых систем.

Вы можете использовать любую файловую систему для первого раздела, но FAT32 является универсальной и позволяет загрузить SystemRescue в Linux, macOS и Windows. Для копирования файлов из macOS и Windows отформатируйте раздел данных с файловой системой FAT32 или exFAT.

Второй раздел вам придется смонтировать вручную, и только после этого вы сможете применять его по своему усмотрению. С его помощью можно копировать файлы из хост-системы, но самое замечательное в наличии раздела, доступного для записи, — это возможность использовать его для хранения изменений в SystemRescue, таких как изменения в конфигурационных файлах или установленное программное обеспечение. См. рецепт 19.14, в котором подробно рассказывается об этом.

Дополнительная информация

- Глава 8.
- Глава 11.
- https://system-rescue.org.
- `man 1 isohybrid`
- `man 1 dd`

19.14. Сохранение изменений в SystemRescue

Задача

SystemRescue прекрасный инструмент, но хотелось бы иметь возможность сохранить некоторые изменения и не начинать каждый сеанс с настроек.

Решение

В рецепте 19.13 рассказывается, как создать доступный для записи раздел на USB-носителе с SystemRescue. Присвойте файловой системе в этом разделе какую-нибудь говорящую метку, например `data`. Затем загрузите SystemRescue, выберите понравившийся вам пункт в загрузочном меню, нажмите клавишу TAB и добавьте параметр **`cow_label=`*****`data`*** к выбранному пункту меню (рис. 19.8).

SystemRescue смонтирует оба ваших раздела в `/run/archiso/`:

```
# lsblk -p
lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
[...]
sdc      8:32   1   3.7G  0 disk
├─sdc1   8:33   1     2G  0 part /run/archiso/bootmnt
└─sdc2   8:34   1   152G  0 part /run/archiso/cowspace
```

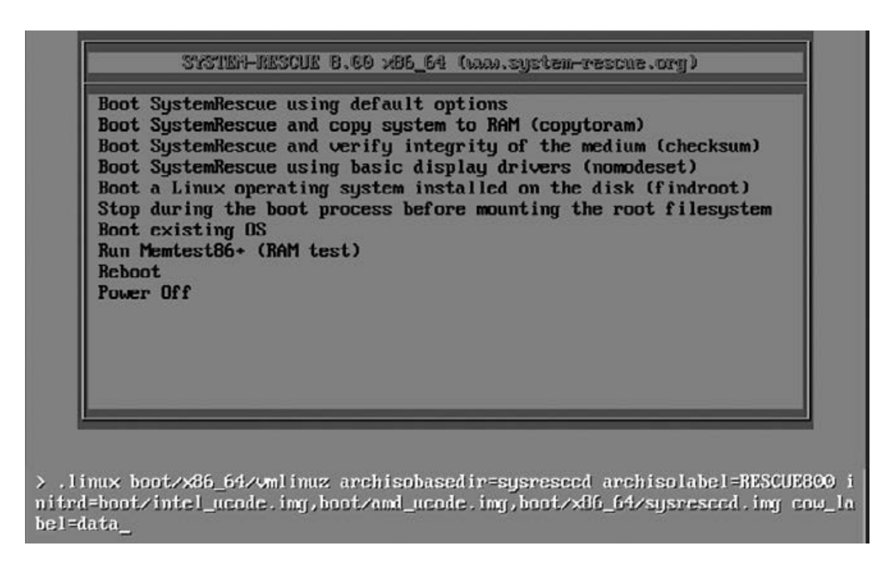


Рис. 19.8. Добавление параметра в пункт загрузочного меню

Все изменения, которые вы вносите в SystemRescue, сохранятся в `/run/archiso/cowspace/persistent_RESCUE800`, а корневая файловая система остается неизменной. Часть имени `RESCUE800` может меняться в зависимости от номера версии SystemRescue.

Вы можете настроить SystemRescue так же, как любую систему Linux: включать и отключать службы, устанавливать пароль root, изменять внешний вид, устанавливать новое программное обеспечение, изменять конфигурационные файлы с настройками сети или создавать новые документы.

Комментарий

USB-носители большого объема недороги, и они удобны в качестве устройства для проведения восстановительных работ. Вы можете использовать флеш-накопители USB или жесткие диски USB.

Дополнительная информация

- https://system-rescue.org.

ГЛАВА 20

Устранение неполадок на компьютере с Linux

Linux включает множество утилит, помогающих диагностировать и устранять проблемы, описание которых легко может занять несколько толстых книг. В этой главе основное внимание уделяется использованию системных журналов для выяснения причин неполадок, созданию центрального сервера журналирования systemd, мониторингу состояния оборудования, поиску и остановке проблемных процессов, обеспечению максимальной производительности оборудования, а также советам и рекомендациям по диагностике проблем с оборудованием.

Обзор

Внимательно изучите системные журналы, и вы найдете причины проблем. Если обнаруженная причина не указывает на решение, то у вас по крайней мере есть информация для обращения за помощью к документации по продукту, в платную службу поддержки или к сообществу.

Ознакомьтесь с документацией для своего дистрибутива Linux, особенно с журналами изменений и примечаниями к выпускам, а также с документацией для серверов и приложений, которые вы используете. Ubuntu, Fedora и openSUSE поддерживают свою документацию в превосходном состоянии и выпускают подробные примечания к выпуску. Кроме того, посетите форумы, вики-страницы и чаты для вашего дистрибутива, серверов и приложений. С любой проблемой, с которой столкнулись вы, почти наверняка пришлось столкнуться кому-то еще.

Профилактика

Большинство ошибок связано с программным обеспечением. Даже оборудование потребительского уровня довольно надежно и в большинстве случаев выходит из строя из-за неправильного использования или возраста. Чаще всего отказывает оборудование с движущимися частями, такими как:

- диски SATA и SCSI;
- вентиляторы процессоров;
- блоки питания;
- вентиляторы в корпусе;
- приводы CD/DVD.

Продлить срок службы оборудования помогут простые меры. Перегрев и нестабильное электропитание — вот главные убийцы электроники. Для долгой и надежной работы компьютеров необходимо хорошее охлаждение, которое обеспечивается использованием специально спроектированных корпусов, поддерживающих надлежащий воздушный поток, радиаторов и вентиляторов для процессоров, а также корпусных вентиляторов, расположенных так, что воздух правильно всасывается в корпус и выталкивается из него. Работающие вентиляторы добавляют шума, но можно приобрести малошумные корпуса, блоки питания и вентиляторы. Периодически пылесосьте внутренности компьютера нестатическим пылесосом и очищайте корпусные фильтры. Если вы предпочитаете использовать сжатый воздух для выдувания пыли, то будьте осторожны с вентиляторами. Если вращать их слишком быстро, то можно повредить подшипники.

Стабилизатор питания обеспечит постоянную защиту от скачков напряжения, а также от радио- и электромагнитных помех. Сетевые фильтры стоят дешевле, но защищают только от скачков напряжения вверх. Но падение напряжения не менее опасно. Стабилизатор питания с лихвой окупится за счет увеличения срока службы и стабильной работы.

Терпение

Терпение — ваш лучший друг при устранении проблем. Этим делом лучше заниматься медленно и систематически:

- почитайте инструкции и убедитесь, что ничего не пропустили и не допустили ошибки;
- проверьте наличие обновлений; часто это помогает решить проблему;

- скопируйте сообщения об ошибках и записи из файлов журнала и попробуйте поискать по ним информацию в Интернете и в уведомлениях о неисправностях;
- что случилось перед ошибкой? Какие действия привели к ошибке? Можно ли ее воспроизвести;
- обратимы ли последние события? Если да, то отменяйте их по очереди в обратном порядке и проверяйте, исчезла ли ошибка. Если отменить сразу все, то можно не обнаружить причину ошибки;
- если ничего не помогает, то перезагрузите компьютер. Это удивительно действенное средство решения проблем; правда, вы можете так и не узнать причину проблемы.

Некоторые графические приложения, такие как замечательный каталогизатор и редактор фотографий digiKam, выводят массу деталей, когда запускаются в терминале, как показано в следующем примере неудачной попытки запуска digiKam:

```
$ digikam
Object::connect: No such signal org::freedesktop::UPower::DeviceAdded(QString)
Object::connect: No such signal org::freedesktop::UPower::DeviceRemoved(QString)
digikam: symbol lookup error: digikam: undefined symbol:
_ZNK11KExiv2Iface14AltLangStrEdit8textEditEv
```

Я даже примерно не представляю, что это означает, но кто-то наверняка знает, поэтому я могу применить данный вывод для поиска в Интернете или составления просьбы о помощи на форумах пользователей digiKam.

Когда вы просите помощи, будьте терпеливы и вежливы. Когда вас попросят дать дополнительную информацию, дайте именно то, что спрашивают. Если вы решили проблему, то поделитесь своим решением и поблагодарите людей, которые помогали вам.

20.1. Поиск полезной информации в файлах журналов

Задача

Происходит нечто непонятное, и нужно выяснить причину. С чего начать?

Решение

Включите журналирование всего и вся, а затем прочитайте файлы журналов. Журналы хранятся в каталоге `/var/log`, а команды `dmesg` и `journalctl` помогут прочитать их. Всеми журналами управляет systemd, используя демон *journald*,

поэтому в выводе `dmesg` и `journalctl` можно увидеть много повторяющейся информации.

Команда `dmesg` читает кольцевой буфер ядра — особое место в памяти, предназначенное для регистрации действий ядра. Запустите `dmesg`, чтобы увидеть все, что произошло при запуске; как работало оборудование после запуска, например, какие устройства USB подключались и отключались; какие сетевые события имели место. Кольцевой буфер ядра имеет фиксированный размер, поэтому новые записи затирают самые старые. Однако на самом деле ничего не теряется, поскольку записи сохраняются в `/var/log/messages`, `/var/log/dmesg` и `journalctl`.

Прочитать журнал `dmesg` можно следующим образом:

```
$ dmesg | less
[    0.000000] microcode: microcode updated early to revision 0x28,
date = 2019-11-12
[    0.000000] Linux version 5.8.0-45-generic (buildd@lcy01-amd64-024) (gcc
(Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0, GNU ld (GNU Binutils for Ubuntu) 2.34)
#51~20.04.1-Ubuntu SMP Tue Feb 23 13:46:31 UTC 2021
(Ubuntu 5.8.0-45.51~20.04.1-generic 5.8.18)
[...]
```

При поиске чего-то конкретного, например истоков проблем с накопителем, используйте команду `grep`:

```
$ dmesg | grep -w sd
[11236.888910] sd 7:0:0:0: [sdd] Attached SCSI removable disk
[11245.095341] FAT-fs (sdd1): Volume was not properly unmounted. Some data may
be corrupt. Please run fsck.
```



Поиск по целому слову с помощью grep

Чтобы выполнить поиск по целому слову, используйте grep с параметром -w. Например, если просто передать команде grep последовательности символов ping, то она вернет записи, включающие такие слова, как piping, escaping, sleeping. Параметр -w позволяет получить совпадения только с целым словом.

Используйте команду `dmesg` с параметром `-T`, чтобы вывести время в удобочитаемом формате:

```
$ dmesg -T | less
[Tue Mar 23 15:25:17 2021] PCI: CLS 64 bytes, default 64
[Tue Mar 23 15:25:17 2021] Trying to unpack rootfs image as initramfs...
[Tue Mar 23 15:25:17 2021] Freeing initrd memory: 56008K
[...]
```

По умолчанию время выводится в секундах и наносекундах, прошедших с момента запуска системы. Для наблюдения за появлением новых событий

по мере их возникновения, например событий подключения и отключения USB-устройств, используйте команду `dmesg --follow`. Для остановки нажмите Ctrl+C.

Ищите записи с определенными уровнями журналирования, такие как ошибки и предупреждения:

```
$ dmesg -l err,warn
```

Выполните команду `dmesg -h`, чтобы получить список доступных подкоманд и параметров.

Каталог `/var/log` — это устаревшее местоположение для хранения файлов журналов, но вы все равно найдете там некоторые журналы в зависимости от того, как ваш дистрибутив Linux управляет ими. Большинство файлов в `/var/log` имеют простой текстовый формат, поэтому легко поддаются поиску. Если вы не знаете, с чего начать поиск, то выполните поиск по всему каталогу с помощью команды `grep`.

Например, вы помните, что установили приложение *graphicsmagick*, но не можете его найти. Загляните в `/var/log`, там есть запись о его установке:

```
$ sudo grep -ir graphicsmagick /var/log
apt/history.log:Install: libgraphicsmagick-q16-3:amd64 (1.4+really1.3.35-1,
automatic), graphicsmagick:amd64 (1.4+really1.3.35-1)
[...]
/var/log/dpkg.log:2021-03-11 17:00:57 install libgraphicsmagick-q16-3:amd64
1.4+really1.3.35-1
[...]
```

Все журналирование в системах с systemd осуществляется через `journalctl`, поэтому можно использовать исключительно данную команду и забыть о `dmesg` и `/var/log`:

```
$ journalctl
```

Вызовите ее с помощью `sudo`, чтобы увидеть дополнительную информацию, которая обычно недоступна рядовым пользователям.

По умолчанию `journalctl` первыми выводит самые старые записи. Нажимайте клавишу пробела или PageUp/PageDown для прокрутки сразу на целый экран или клавиши со стрелками для прокрутки на одну строку. Нажатие комбинации Ctrl+End перенесет вас в самый конец журнала, а Ctrl+Home вернет в начало. Для выхода нажмите клавишу Q.

Для начала просмотрите самые новые записи:

```
$ journalctl -r
```

По умолчанию длинные строки не переносятся, поэтому для их чтения необходимо использовать клавиши со стрелками. Добавьте перенос, передав вывод `journalctl` команде `less`:

```
$ journalctl -r | less
```

Просмотрите самые последние записи с пояснительными сообщениями, если таковые имеются. Ниже представлен пример такого сообщения с пояснением:

```
$ journalctl -ex | less

-- The unit grub-initrd-fallback.service has successfully entered the 'dead'
state.
Mar 27 10:14:29 client4 systemd[1]: Finished GRUB failed boot detection.
-- Subject: A start job for unit grub-initrd-fallback.service has finished
successfully
-- Defined-By: systemd
```

Ищите конкретные службы, такие как MariaDB:

```
$ sudo journalctl -u mariadb.service
Mar 19 16:07:27 client4 /etc/mysql/debian-start[7927]: Looking for 'mysql' as:
/usr/bin/mysql
Mar 19 16:07:27 client4 /etc/mysql/debian-start[7927]: Looking for 'mysqlcheck'
as: /usr/bin/mysqlcheck
[...]
```

Ограничьте поиск диапазоном дат. Это можно сделать несколькими способами:

```
$ journalctl -u mariadb.service -S today
$ journalctl -u ssh.service -S '1 week ago'
$ journalctl -u libvirtd.service -S '2021-03-05'
$ journalctl -u httpd.service -S '2021-03-05' -u '2021-03-09'
$ journalctl -u nginx.service -S '2 hours ago'
```

Если время не задано явно, то по умолчанию подразумевается 00:00:00, полночь. Задавайте время в формате ЧЧ:ММ:СС:

```
$ journalctl -u httpd.service -S '2021-03-05 13:15:00' -U now
```

Просмотреть события, имевшие место от часа до пяти минут тому назад, и имена файлов модулей, ответственных за эти события, можно следующим образом:

```
$ journalctl -S '1h ago' -U '5 min ago' -o with-unit
```

Команда `journalctl` сортирует журналы по загрузке системы. Просмотреть события сервера HTTP, произошедшие с момента последней загрузки, и вывести при этом только 50 самых последних записей можно следующим образом:

```
$ journalctl -b -n 50 -u httpd.service
```

Посмотреть, что происходило три загрузки тому назад, можно так:

```
$ journalctl -b -2 -u httpd.service
```

Список всех загрузок с отметками времени можно получить следующим образом:

```
$ journalctl --list-boots
```

Есть возможность фильтровать записи по уровням серьезности. Когда указывается один уровень, как в этом примере, `crit`, то будут показаны все сообщения с уровнем `crit` и выше, вплоть до самого серьезного уровня `emerg`:

```
$ journalctl -b -1 -p "crit" -u nginx.service
```

Можно определить диапазон уровней серьезности, например, от `crit` до `warning`:

```
$ journalctl -b -3 -p "crit".."warning"
```

Наблюдать за появлением новых событий с предварительным выводом десяти предыдущих последних событий можно так:

```
$ journalctl -n 10 -u mariadb.service -f
```

Нажмите Ctrl+C, чтобы остановить наблюдение.

И конечно, используйте старую добрую команду `grep` для поиска, например, имен пользователей или любых других последовательностей символов:

```
$ journalctl -b -1 | grep madmax
```

Комментарий

Уровни серьезности — это стандартные значения от 0 до 7 для syslog, где уровень 0 считается наиболее серьезным, а уровень 7 — наименее:

```
emerg      (0)
alert      (1)
crit       (2)
err        (3)
warning    (4)
notice     (5)
info       (6)
debug      (7)
```

Команда `journalctl` поддерживает десятки способов анализа и фильтрации вывода, и все они описываются на странице справочного руководства `man 1 journalctl`.

Дополнительная информация

- `man 3 syslog`
- `man 1 journalctl`
- `man 1 dmesg`
- https://systemd.io.

20.2. Настройка демона journald

Задача

Узнать, какие настройки *journald* используются в данный момент, и при необходимости изменить их.

Решение

Настройки journald находятся в `/etc/systemd/journald.conf`. Некоторые параметры по умолчанию закомментированы, а все значения по умолчанию, установленные во время компиляции, задокументированы в руководстве `man 5 journald.conf`. Далее мы рассмотрим параметры, которые наиболее часто приходится настраивать.

`Storage=auto` по-разному интерпретируется в разных дистрибутивах. В Ubuntu и Fedora энергозависимое хранилище находится в `/run/log/journal/`, а постоянное — в `/var/log/journal`. С помощью команды `systemctl` можно узнать местоположение файлов журнала, а также занимаемое и свободное пространство:

```
$ systemctl status systemd-journald.services
● systemd-journald.service - Journal Service
   Loaded: loaded (/usr/lib/systemd/system/systemd-journald.service; static;
   vendor preset: disabled)
   Active: active (running)
   [...]
Mar 27 15:04:40 server2 systemd-journald[508]: Runtime journal (/run/log/
journal/
1181e27c52294e97a8ca5c5af5c92e20) is 8.0M, max 2.3G, 2.3G free.
Mar 27 15:04:55 server2 systemd-journald[508]: Time spent on flushing to /var is
381.408ms for 1176 entries.
Mar 27 15:04:55 server2 systemd-journald[508]: System journal (/var/log/journal/
1181e27c52294e97a8ca5c5af5c92e20) is 16.0M, max 4.0G, 3.9G free.
```

В openSUSE энергозависимое хранилище находится в `/run/log/journal/`, а постоянное — в `/var/log/messages`. Если вы предпочитаете использовать

/var/log/journal, то создайте этот каталог и измените группу-владельца на systemd-journal:

```
$ sudo mkdir /var/log/journal
$ sudo chgrp /var/log/journal/ systemd-journal
```

Больше ничего менять не нужно, и после перезагрузки для хранения журналов будет выбран ваш каталог. Другие возможные значения этого параметра: volatile, persistent и none:

- volatile — журналы сохраняются только в оперативной памяти, в /run/log/journal/;
- persistent — журналы сохраняются на диске, а хранилище в /run/log/journal/ используется, только если диск недоступен, например на первых этапах загрузки системы;
- none — запрещает локальное журналирование и у вас есть возможность отправлять сообщения на центральный сервер журналирования.

Параметр SystemMaxUse= задает размер хранилища журналов на диске, а RuntimeMaxUse= — размер энергозависимого хранилища. По умолчанию размер хранилища составляет 10 % от доступного пространства в файловой системе, но не более 4 Гбайт.

Параметры SystemKeepFree= и RuntimeKeepFree= задают размер дискового пространства, которое должно оставаться свободным для других целей. По умолчанию 15 % и 4 Гбайт. Значения этих параметров можно указать в байтах или использовать единицы измерения K, M, G, T, P и E; например, 25 G (гигабайт).

Параметр MaxRetentionSec= определяет продолжительность хранения файлов. Значение по умолчанию — 0, то есть продолжительность хранения файлов определяется другими настройками, такими как доступное дисковое пространство. Значения времени задаются в секундах, но можно использовать и единицы измерения year (год), month (месяц), week (неделя), day (день), h (часы) или m (минуты), например, 6 month.

Комментарий

Демон journald автоматически выполняет ротацию журналов. Активные файлы преобразуются в архивные, а архивные файлы удаляются в соответствии с настройками.

Дополнительная информация

- man 5 journald.conf

20.3. Создание сервера журналирования с помощью systemd

Задача

Настроить центральный сервер журналирования, чтобы журналы сохранялись при выходе системы из строя, а также чтобы централизованно управлять ими.

Решение

В дистрибутивах с systemd имеется демон удаленного журналирования journald. Клиентские машины отправляют свои сообщения на сервер journald. Для организации такого сервера потребуются:

- компьютер для хранения файлов журналов;
- доступность сервера журналирования для клиентов по сети;
- пакет *systemd-journal-remote*, установленный на сервере и на всех клиентах;
- настроенная инфраструктура открытых ключей (Public Key Infrastructure, PKI), как описывалось в рецепте 13.5, с ключами и сертификатами, распространенными по серверам и клиентам.

Когда systemd-journal-remote будет установлен, отредактируйте `/etc/systemd/journal-remote.conf` на сервере. Я предпочитаю хранить ключи шифрования и сертификаты в `/etc/pki/journald/`:

```
[Remote]
Seal=false
SplitMode=host
ServerKeyFile=/etc/pki/journald/log-server.key
ServerCertificateFile=/etc/pki/journald/log-server.crt
TrustedCertificateFile=/etc/pki/journald/ca.crt
```

Установите разрешения на доступ к ключам и сертификатам сервера:

```
$ sudo chmod -R 0755 /etc/pki/journald
$ sudo chmod 0440 /etc/pki/journald/log-server.key
```

Измените группу-владельца закрытого ключа сервера на systemd-journal-remote:

```
$ sudo chgrp systemd-journal-remote /etc/pki/journald/log-server.key
```

Включите и запустите службу systemd-journal-remote, предварительно запустив systemd-journalremote.socket:

```
$ sudo systemctl enable --now systemd-journal-remote.socket
$ sudo systemctl enable --now systemd-journal-remote.service
```

Проверьте состояние обеих служб и убедитесь, что они запустились корректно. Откройте необходимые порты в брандмауэре на сервере:

```
$ sudo firewall-cmd --zone=internal --add-port=19532/tcp
$ sudo firewall-cmd --zone=internal --add-port=80/tcp
$ sudo firewall-cmd --runtime-to-permanent
$ sudo firewall-cmd --reload
```

На каждом клиенте создайте нового пользователя systemd-journal-upload. Эту учетную запись будет использовать процесс systemd-journal-upload для передачи журналируемых сообщений на центральный сервер:

```
$ sudo useradd -r -d /run/systemd -M -s /usr/sbin/nologin -U \
systemd-journal-upload
```

Настройте разрешения для ключей и сертификатов на клиентах:

```
$ sudo chmod -R 0755 /etc/pki/journald
$ sudo chmod 0440 /etc/pki/journald/client.key
```

Отредактируйте `/etc/systemd/journal-upload.conf`, указав URL порт TCP сервера журналирования, а также местоположение клиентских ключей и сертификатов:

```
[Upload]
URL=https://logserver.example.com:19532
ServerKeyFile=/etc/pki/journald/client1.key
ServerCertificateFile=/etc/pki/journald/client1.crt
TrustedCertificateFile=/etc/pki/journald/ca.crt
```

Перезапустите службу systemd-journal-upload.service:

```
$ sudo systemctl restart systemd-journal-upload.service
```

Если перезапуск прошел успешно, без ошибок, то выполните следующие действия, чтобы проверить, что клиент отправляет записи на сервер. Проверьте каталог для хранения журналов на сервере:

```
$ sudo ls -la /var/log/journal/remote/
total 7204
```

```
drwxr-xr-x  2 systemd-journal-remote systemd-journal-remote  6 Mar 26 16:41 .
drwxr-sr-x+ 4 root                   systemd-journal        60 Mar 26 16:41 ..
rw-r-----  1 systemd-journal-remote systemd-journal-remote 8388608 Mar 26  1
10:46 'remote-CN=client1.example.com'
```

Пока все идет неплохо. Теперь отправьте серверу сообщение с клиента:

```
$ sudo logger -p syslog.debug "Hello, I am client1! Do you hear me?"
```

Выполните команду `journalctl` на сервере, чтобы увидеть самые последние записи. Если вы увидели сообщение клиента, значит, все настроено правильно:

```
Mar 27 18:30:11 client1 madmax[15228]: Hello, I am client1! Do you hear me?
```

Комментарий

Центральный сервер журналирования выступает единым хранилищем для журналов клиентов, упрощая их обслуживание и анализ. Каждому клиенту на сервере отводится свой каталог.

Параметр `Seal=false` отключает добавление криптографической подписи к записям в журнале. Если вы решите опробовать эту возможность, то прочитайте описание параметра `--setup-keys` в руководстве: `man 1 journalctl`. Я не смогла найти однозначного ответа на вопрос о полезности данной возможности, но знание о ее наличии вам не повредит.

Параметр `SplitMode=host` обеспечивает хранение журнала каждого клиента в отдельном файле. Если установить значение `false`, то все журналы будут сохраняться в одном файле.

Параметры `ServerKeyFile=`, `ServerCertificateFile=` и `TrustedCertificateFile=` описывают местоположение ключей шифрования и сертификатов.

Дополнительная информация

- `man 5 journal-remote.conf`
- `man 5 journald.conf`
- `man 1 journalctl`

20.4. Мониторинг температуры, частоты вращения вентиляторов и уровня напряжения с помощью lm-sensors

Задача

Измерить температуру внутри корпуса компьютера, частоту вращения вентиляторов и уровни напряжения.

Решение

Постоянный мониторинг температуры процессора, жесткого диска и корпуса можно проводить с помощью программного обеспечения *lm-sensors* для чтения показаний датчиков. В openSUSE оно входит в состав пакета *sensors*, в Fedora — *lm_sensors*, и в Ubuntu — *lm-sensor*.

Когда lm-sensors будет установлено, запустите команду `sensors-detect`, чтобы откалибровать lm-sensors для вашего оборудования:

```
$ sudo sensors-detect
# sensors-detect version 3.6.0
# Board: ASRock H97M Pro4
# Kernel: 5.8.0-45-generic x86_64
# Processor: Intel(R) Core(TM) i7-4770K CPU @ 3.50GHz (6/60/3)

This program will help you determine which kernel modules you need to load to
use lm_sensors most effectively. It is generally safe and recommended to accept
the default answers to all questions, unless you know what you're doing.

Some south bridges, CPUs or memory controllers contain embedded sensors.
Do you want to scan for them? This is totally safe. (YES/no):
[...]
```

Нажимайте Enter, чтобы принять ответы по умолчанию. По завершении вы увидите примерно такое сообщение:

```
To load everything that is needed, add this to /etc/modules:
#----cut here----
# Chip drivers
coretemp
nct6775
```

```
#----cut here----
If you have some drivers built into your kernel, the list above will
contain too many modules. Skip the appropriate ones!
Do you want to add these lines automatically to /etc/modules? (yes/NO) yes
Successful!
```

Модули будут загружены автоматически после перезагрузки системы. Но их можно загрузить и немедленно:

```
$ sudo systemctl restart systemd-modules-load.service
```

Теперь запустите команду `sensors` и посмотрите на результат:

```
$ sensors
coretemp-isa-0000
Adapter: ISA adapter
Package id 0:  +42.0°C  (high = +86.0°C, crit = +96.0°C)
Core 0:        +34.0°C  (high = +86.0°C, crit = +96.0°C)
Core 1:        +35.0°C  (high = +86.0°C, crit = +96.0°C)
Core 2:        +32.0°C  (high = +86.0°C, crit = +96.0°C)
Core 3:        +31.0°C  (high = +86.0°C, crit = +96.0°C)

nouveau-pci-0300
Adapter: PCI adapter
GPU core:     +1.01 V  (min =  +0.70 V, max =  +1.20 V)
fan1:        2850 RPM
temp1:        +51.0°C  (high = +95.0°C, hyst =  +3.0°C)
                       (crit = +105.0°C, hyst =  +5.0°C)
                       (emerg = +135.0°C, hyst =  +5.0°C)

dell_smm-virtual-0
Adapter: Virtual device
Processor Fan: 1070 RPM
Other Fan:        0 RPM
Other Fan:      603 RPM
CPU:            +41.0°C
SODIMM:         +25.0°C
SODIMM:         +35.0°C
SODIMM:         +34.0°C
```

Здесь отображается информация о ядрах процессора, графическом адаптере, вентиляторах и модулях памяти. Кроме текущих температур, выводятся также диапазоны высоких, критических и аварийных температур. Процессоры имеют встроенную защиту от перегрева и автоматически отключаются, когда температура становится слишком высокой.

Для просмотра текущих значений с обновлением каждые две секунды и выделением любых различий используйте команду `watch`:

```
$ watch -d sensors
```

При желании интервал обновления можно изменить, например, установить равным 10 секундам:

```
$ watch -d -n 10 sensors
Every 10.0s: sensors
[...]
```

Для остановки нажмите комбинацию Ctrl+C.

Комментарий

В lm_sensors нет никакого волшебства. Это программное обеспечение просто читает показания имеющихся аппаратных датчиков, для которых имеются драйверы Linux. Большинство аппаратных датчиков имеют невысокую точность, поэтому пусть вас не беспокоят небольшие колебания.

Мониторинг температуры, напряжения и частоты вращения вентилятора может заранее предупредить о проблемах. Заменить вентилятор дешевле, чем восстановить сгоревший компьютер. Падение напряжения может указывать на сбои в источнике питания или плохой контакт.

Прежде чем вносить изменения в файл `/etc/modules`, проверьте конфигурацию ядра — возможно, какие-то из модулей, предлагаемых утилитой sensor-detect, скомпилированы статически с ядром. Файл конфигурации ядра находится в каталоге `/boot` и имеет имя `config-<`*`версия-ядра`*`>`, например `config-5.8.0-45-generic`. Попробуем найти модуль `nct6775`:

```
$ grep -i nct6775 config-5.8.0-45-generic
CONFIG_SENSORS_NCT6775=m
```

Значение `m` говорит о том, что драйвер скомпилирован как загружаемый модуль ядра. Проверим, возможно, он уже загружен:

```
$ lsmod | grep nct6775
```

Если эта команда ничего не вернет, то можете смело добавить модуль в `/etc/modules`. Если драйвер статически скомпилирован с ядром, то в `config-*` будет найдена примерно такая строка:

```
CONFIG_SENSORS_NCT6775=y
```

Значение `y` означает, что драйвер встроен в ядро, поэтому его не нужно добавлять в `/etc/modules`.

Дополнительная информация

- `man 1 watch`
- `man 1 sensors`
- `man 8 lsmod`
- https://kernel.org.

20.5. Добавление графического интерфейса для lm-sensors

Задача

Получить графический интерфейс для lm-sensors, обновляющийся автоматически.

Решение

На выбор есть несколько хороших вариантов. Графические интерфейсы для lm-sensors также поддерживают другие инструменты мониторинга, такие как *smartmontools* и *hddtemp*. Psensor, например, предлагает большую графическую панель с цветными графиками и простыми настройками для изменения надписей и отображения только нужной вам информации (рис. 20.1).

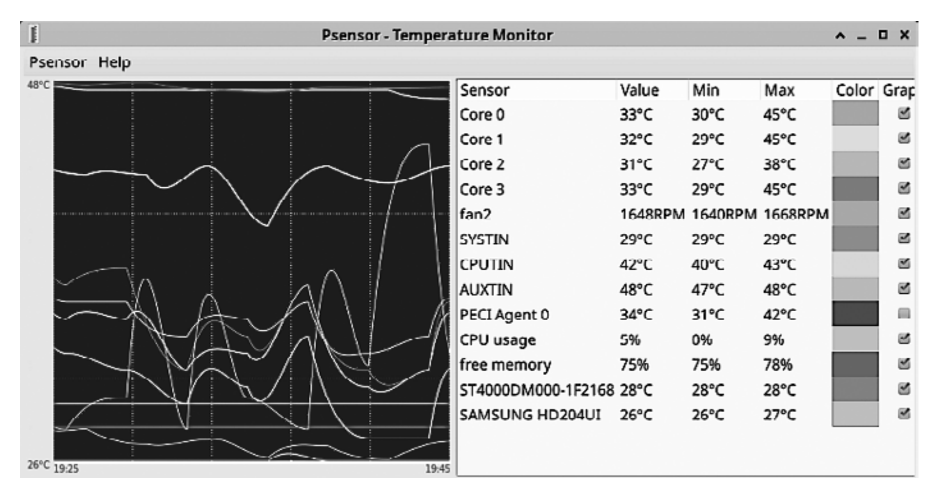


Рис. 20.1. Psensor следит за изменениями показаний аппаратных датчиков

Psensor имеет возможность оповещать о превышении пороговых значений. Оповещение включается отдельно для каждого сигнала, таких как потребление процессора или частота вращения вентилятора. Просто щелкните на выбранном датчике, чтобы открыть диалог с настройками (рис. 20.2).

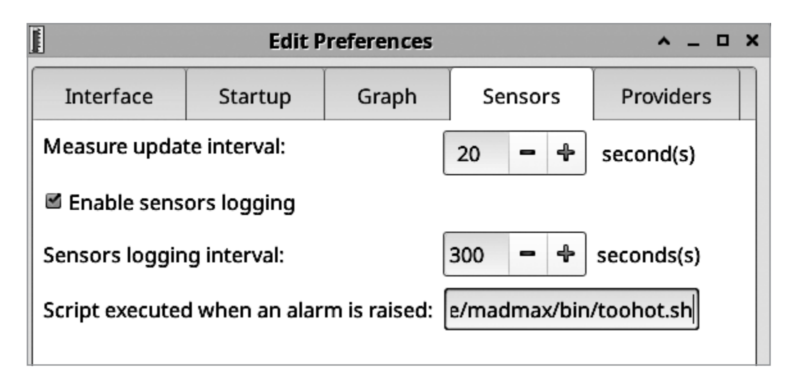


Рис. 20.2. Флажок включения уведомления и настройки интервалов проверки и журналирования

Для настройки уведомлений нужно написать короткий сценарий, как в примере ниже:

```
#!/bin/bash
# toohot.sh, проигрывает умопомрачительную музыкальную фразу, когда
# показания датчика превышают верхний предел

play /home/madmax/Music/klavichord-4.wav
```

Установите пакет *sox*, чтобы получить возможность использовать команду `play`. Сделайте свой сценарий выполняемым:

```
$ chmod +x toohot.sh
```

Протестируйте его:

```
$ play toohot.sh
```

Добившись желаемого звучания, укажите его в настройках Psensor. Откройте диалог Psensor ▸ Preferences ▸ Sensors (Psensor ▸ Параметры ▸ Датчики) (рис. 20.3).

Самый простой способ проверить сценарий в Psensor — установить слишком низкую максимальную температуру.

Многие среды рабочего стола, такие как Xfce4, GNOME и KDE, имеют небольшие плагины для отображения на панели задач, как, например, плагины из Xfce4, изображенные на рис. 20.4.

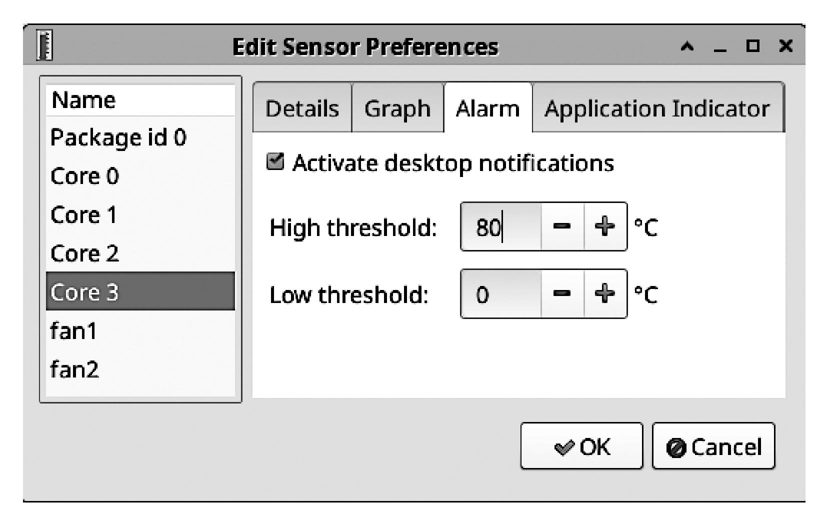


Рис. 20.3. Настройка уведомлений

CPU: 36 °C fan2: 1646 rpm
Motherboard: 32 °C Power Supply: 44 °C

Рис. 20.4. Плагин для панели задач в Xfce, отображающий датчики lm-sensors

Все они имеют в названиях пакетов слово sensor, кроме *gnome-shell-extension-freon*.

Комментарий

Вы можете предусмотреть в своем сценарии автоматическое выключение системы при срабатывании датчика, как в следующем простом примере:

```
#!/bin/bash
echo "Помогите, слишком горячо! Я выключусь прямо сейчас!" && shutdown -h now
```

Дополнительная информация

- `man 1 play`
- `man 1 sensors`
- Psensor (https://oreil.ly/IcRok).

20.6. Мониторинг состояния жесткого диска с помощью smartmontools

Задача

Организовать мониторинг состояния жесткого диска, чтобы не пропустить момент, когда он выйдет из строя или, что предпочтительнее, когда он будет на грани выхода из строя, чтобы успеть заменить его и не потерять данные.

Решение

Большинство жестких дисков и твердотельных накопителей поддерживают технологию S.M.A.R.T. (Self-Monitoring Analysis and Reporting Technology — технология самоконтроля, самоанализа и составления диагностических отчетов). С ее помощью они отслеживают и фиксируют определенные характеристики работы, на основании которых можно (хотелось бы надеяться) предсказать сбои. В Linux есть набор инструментов *smartmontools* для чтения этой информации и выдачи предупреждений.

Инструменты smartmontools распространяются в составе одноименного пакета. После его установки автоматически должна установиться и запуститься служба systemd, в чем можно убедиться с помощью команды `systemctl`:

```
$ systemctl status smartd.service
```

Используйте команду `smartctl`, чтобы узнать, поддерживает ли ваш диск технологию S.M.A.R.T. Ищите строки со словом `SMART`:

```
$ sudo smartctl -i /dev/sda
smartctl 7.1 2019-12-30 r5022 [x86_64-linux-5.8.0-45-generic] (local build)
Copyright (C) 2002-19, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF INFORMATION SECTION ===
Model Family:     Seagate Desktop HDD.15
Device Model:     ST4000DM000-1F2168
[...]
SMART support is: Available - device has SMART capability.
SMART support is: Enabled
```

Включите или выключите команду `smartctl` для каждого диска:

```
$ sudo smartctl -s on /dev/sda
$ sudo smartctl -s off /dev/sda
```

Добавьте параметр `-x` для вывода полного объема информации:

```
$ sudo smartctl -x /dev/sda
```

Выполните быструю проверку состояния:

```
$ sudo smartctl -H /dev/sda
smartctl 7.1 2019-12-30 r5022 [x86_64-linux-5.8.0-45-generic] (local build)
Copyright (C) 2002-19, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED
```

Используйте комбинацию флагов `-Hc`, чтобы получить полный отчет.

Загляните в файл журнала:

```
$ sudo smartctl -l error /dev/sda
smartctl 7.0 2019-05-21 r4917 [x86_64-linux-5.3.18-lp152.66-preempt] (SUSE RPM)
Copyright (C) 2002-18, Bruce Allen, Christian Franke, www.smartmontools.org

=== START OF READ SMART DATA SECTION ===
SMART Error Log Version: 1
No Errors Logged
```

Есть два вида самопроверки — быстрая и детальная. В момент запуска они сообщат вам, сколько времени займет каждая из них:

```
$ sudo smartctl -t long /dev/sda
[...]
=== START OF OFFLINE IMMEDIATE AND SELF-TEST SECTION ===
Sending command: "Execute SMART Extended self-test routine immediately in
off-line mode".
Drive command "Execute SMART Extended self-test routine immediately in off-line
mode" successful.
Testing has begun.
Please wait 109 minutes for test to complete.
Test will complete after Thu Mar 25 17:06:33 2021

Use smartctl -X to abort test.
```

Они не уведомляют о завершении, но вы в любое время можете проверить файл журнала:

```
$ sudo smartctl -l selftest /dev/sda

[sudo] password for carla:
[...]
=== START OF READ SMART DATA SECTION ===
SMART Self-test log structure revision number 1
Num  Test_Description    Status                  Remaining  LifeTime(hours)
```

```
# 1  Extended offline    Self-test routine in progress 70%      7961
# 2  Short offline       Completed without error       00%      7960
# 3  Short offline       Completed without error       00%      7952
[...]
```

Не забывайте периодически обновлять базу данных с информацией о жестких дисках:

```
$ sudo update-smart-drivedb
/usr/share/smartmontools/drivedb.h updated from branches/RELEASE_7_0_DRIVEDB
```

Комментарий

Технология S.M.A.R.T. обеспечивает надежность на уровне около 60 %. Могло быть и лучше, но стандарт S.M.A.R.T. оставляет слишком много места для интерпретации, и каждый производитель жестких дисков реализует его по-своему. Документация производителей не отличается богатством информации, и лучший ресурс, по моему мнению, — это «Википедия» и сайт проекта https://smartmontools.org. Как всегда, ваша лучшая страховка — регулярное резервное копирование.

И все же это бесплатный, простой в применении и часто весьма полезный набор инструментов. Обратите внимание на атрибуты с метками `Pre-fail` (которые можно видеть в следующем фрагменте) и прочитайте еще раз введение к данной главе, где говорится о том, как повысить надежность и безотказность ваших систем.

Запустите команду `sudo smartctl -a /dev/sda`, чтобы получить всю имеющуюся информацию S.M.A.R.T. Ниже представлен раздел, заслуживающий особого внимания:

```
SMART Attributes Data Structure revision number: 10
Vendor Specific SMART Attributes with Thresholds:
ID# ATTRIBUTE_NAME          FLAG     VALUE WORST THRESH TYPE      UPDATED
  1 Raw_Read_Error_Rate     0x000f   119   099   006    Pre-fail  Always
  3 Spin_Up_Time            0x0003   092   091   000    Pre-fail  Always
  4 Start_Stop_Count        0x0032   099   099   020    Old_age   Always
  5 Reallocated_Sector_Ct   0x0033   100   100   010    Pre-fail  Always
  7 Seek_Error_Rate         0x000f   059   057   030    Pre-fail  Always
  9 Power_On_Hours          0x0032   089   089   000    Old_age   Always
 10 Spin_Retry_Count        0x0013   100   100   097    Pre-fail  Always
 12 Power_Cycle_Count       0x0032   099   099   020    Old_age   Always
183 Runtime_Bad_Block       0x0032   100   100   000    Old_age   Always
184 End-to-End_Error        0x0032   100   100   099    Old_age   Always
187 Reported_Uncorrect      0x0032   100   100   000    Old_age   Always
188 Command_Timeout         0x0032   100   099   000    Old_age   Always
189 High_Fly_Writes         0x003a   100   100   000    Old_age   Always
190 Airflow_Temperature_Cel 0x0022   072   059   045    Old_age   Always
```

```
191 G-Sense_Error_Rate      0x0032   100   100   000    Old_age   Always
192 Power-Off_Retract_Count 0x0032   100   100   000    Old_age   Always
193 Load_Cycle_Count        0x0032   096   096   000    Old_age   Always
194 Temperature_Celsius     0x0022   028   041   000    Old_age   Always
197 Current_Pending_Sector  0x0012   100   100   000    Old_age   Always
198 Offline_Uncorrectable   0x0010   100   100   000    Old_age   Offline
199 UDMA_CRC_Error_Count    0x003e   200   200   000    Old_age   Always
240 Head_Flying_Hours       0x0000   100   253   000    Old_age   Offline
241 Total_LBAs_Written      0x0000   100   253   000    Old_age   Offline
242 Total_LBAs_Read         0x0000   100   253   000    Old_age   Offline
```

В столбце TYPE выводится тип атрибута: Pre-fail или Old_age. Атрибуты Pre-fail и Old_age не означают, что диск обречен, они просто отражают тип атрибута в этой строке.

- Pre-fail — критический атрибут, который может указывать на неизбежный отказ; он всегда включается в оценку работоспособности.
- Old_age — некритический атрибут; он не включается в отчеты с информацией о работоспособности дисков.

В столбцах ID# и ATTRIBUTE_NAME идентифицируется каждый атрибут. Они зависят от производителя.

В столбце FLAG показан флаг обработки атрибута, не имеющий отношения к работоспособности диска.

В столбце VALUE отображаются текущие значения атрибутов. Диапазон значений — от 0 до 255, за исключением значений 0, 254 и 255. Значение 253 означает «неиспользованный», характерно для новых дисков. Значение в столбце VALUE отражает шкалу значений от хорошего к плохому — чем выше значение, тем ближе атрибут к оценке «хорошо», чем меньше, тем ближе к оценке «плохо», за исключением атрибутов со значениями температуры, которые просто отражают температуру в градусах Цельсия.

В столбце WORST указывается наименьшее значение, зафиксированное для этого атрибута.

В столбце THRESH отображается самый низкий порог для каждого атрибута. Когда значение атрибута с типом Pre-fail опускается ниже значения в столбце THRESH, сбой диска может быть неизбежен.

В столбце UPDATED указывается, когда проверяется значение атрибута. Атрибуты, помеченные как Always, проверяются всегда — при быстрых и детальных проверках. Атрибуты, помеченные как Offline, обычно проверяются только при детальных проверках. Но метки в этом столбце не всегда точно отражают действительность и в любом случае они не особенно полезны.

Если атрибут переходит в состояние сбоя, время сбоя фиксируется в столбце WHEN_FAILED.

Значение RAW_VALUE определяется индивидуально каждым производителем. Просто игнорируйте его.

Дополнительная информация

- man 8 smartctl
- man 8 smartd
- man 8 update-smart-drivedb
- man 5 smartd.conf

20.7. Настройка smartmontools для отправки отчетов по электронной почте

Задача

Организовать отправку уведомлений по электронной почте о любых проблемах, выявленных smartd.

Решение

Сначала проверьте, настроена ли и работает ли в вашей системе почта, отправив тестовое сообщение другому пользователю системы, например root:

```
$ echo "Hello, this is my message" | mail -s "Message subject" root@localhost
```

```
[root@localhost ~]# mail
"/var/mail/root": 1 message 1 unread
>U "/var/mail/root": 1 message 1 new
>N   1 stash    Mon Mar 29 15:26  13/429   Message subject
?
```

Нажмите 1, чтобы прочитать сообщение, и q, чтобы выйти. Данный пример показывает, что почта уже настроена. Если это не так, установите *mailx* и *postfix*. Агент mailx — пользовательский почтовый агент (Mail User Agent, MUA), то есть почтовый клиент, такой же как Evolution, Thunderbird, KMail, Mutt и т. д. А postfix — это агент передачи почты (Mail Transfer Agent, MTA).

Вам понадобятся оба. После установки проверьте с помощью команды `systemctl` — запустились ли они:

```
$ systemctl status smartd.service
$ systemctl status postfix.service
```

Если нет, то включите их и запустите. Затем опять попробуйте послать тестовое сообщение.

```
$ sudo systemctl enable --now smartd.service
$ sudo systemctl enable --now postfix.service
```

Настройки smartd хранятся в файле `/etc/smartd.conf` или `/etc/smartmontools/smartd.conf`. По умолчанию выполняется сканирование всех возможных устройств и отправка по электронной почте отчетов об ошибках пользователю root. Желательно ограничить устройства для мониторинга. В каждом дистрибутиве Linux определена своя особая конфигурация. Следующие настройки должны работать во всех дистрибутивах, и, конечно же, вы должны указать собственные диски и адрес электронной почты:

```
DEFAULT -a -o on -S on -s (S/../.././02|L/../../5/01):
/dev/sda
/dev/sdb
/dev/sdc
DEFAULT -H -m root -M test
```

Сохраните конфигурацию и перезапустите smartd.service:

```
$ sudo systemctl reload smartd.service
```

Комментарий

По умолчанию `smartd.conf` сканирует все доступные диски. Но намного рациональнее указать конкретные диски для мониторинга.

Флаг `-a` объединяет в себе все следующие флаги:

- `-H` проверяет состояние работоспособности, определяемое технологией S.M.A.R.T.;
- `-f` сообщает об атрибутах с типом `Old_age` в состоянии «близкий к отказу» (по значениям `VALUE` и `WORST`);
- `-t` сообщает об изменениях в атрибутах `Pre-fail` и `Old_age`;
- `-l` сообщает об увеличении ошибок ATA;
- `-l selftest` сообщает об увеличении ошибок самоконтроля;
- `-l selfteststs` сообщает об изменении состояний атрибутов, выявленных при проведении самоконтроля;

- `-C 197` сообщает о ненулевом текущем количестве секторов, ожидающих переназначения;
- `-U 198` сообщает о ненулевом текущем количестве сбойных секторов, ожидающих переназначения.

Этот список охватывает наиболее важные аспекты, но вы, конечно же, можете настроить отправку сообщений о любых атрибутах по своему выбору.

Параметр `-M test` будет отправлять указанному пользователю (в предыдущем примере `-m root@localhost`) тестовое сообщение при каждом запуске. Вы можете убрать этот параметр, если уверены, что все работает как должно.

Есть несколько пакетов, включающих двоичный файл *mail*. Вот некоторые из них: *mailutils*, *mailx*, *bsd-mailx* и *s-nail*. Для передачи почты в рамках одной систем подойдет любой из них, и параметры запуска двоичного файла *mail* во всех них одинаковые.

Не обязательно использовать агент postfix — можно задействовать любой MTA, например Exim или Sendmail.

Дополнительная информация

- `man 8 smartd`
- `man 5 smartdconf`

20.8. Диагностика вяло реагирующей системы с помощью команды top

Задача

Ваша система, обычно работающая без нареканий, вдруг стала тормозить и зависать. Приложения запускаются и завершаются слишком долго или вяло реагируют на действия пользователя. Вам нужно выяснить причину такого поведения, а затем устранить ее.

Решение

Запустите команду `top` и посмотрите, какие процессы потребляют слишком много системных ресурсов. Из-за проблем с чрезмерным потреблением процессора и памяти ваша прекрасная мощная система действует ничуть не лучше какой-нибудь древней развалины:

```
$ top
Tasks: 284 total,   1 running, 283 sleeping,   0 stopped,   0 zombie
%Cpu(s):  6.4 us,  4.8 sy,  0.0 ni, 88.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :  15691.4 total,   6758.9 free,   4913.0 used,   4019.6 buff/cache
MiB Swap:  15258.0 total,  15258.0 free,      0.0 used.  10016.5 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   1299 duchess    9   0 2803912  22296  17904 S  80.5   0.1  172:25 Web Content
   1685 duchess   20   0 3756840 543124 241296 S   7.6   3.4   27:53 firefox
  15926 libvirt+  20   0 5151504   2.3g  25024 S   1.7  15.3    1:39 qemu
  [...]
```

Команда top продолжает работать, пока вы не остановите ее, обновляя показания каждые несколько секунд и отображая процессы в порядке от наиболее к наименее активным. Чтобы остановить top, нажмите клавишу q.

Эта команда отображает большой объем информации. Обратите внимание, что в этом примере Web Content потребляет примерно 80,5 % процессорного времени. Плохо продуманные сайты — частая причина падения производительности всей системы. Самый быстрый способ исправить проблему — остановить процесс, являющийся ее причиной.

Идентификаторы процессов выводятся в левом столбце. Нажмите клавишу K, чтобы открыть диалоговое окно остановки процессов. Если идентификатор процесса для остановки определился правильно, то нажмите клавишу Enter. Затем снова нажмите Enter, если вы согласны отправить процессу сигнал по умолчанию 15/sigterm:

```
PID to signal/kill [default pid = 1299]
Send pid 1299 signal [15/sigterm]
```

Если процесс не остановится, то используйте «ядерную дубинку» — сигнал 9/sigkill:

```
PID to signal/kill [default pid = 1299]
Send pid 1299 signal [15/sigterm] 9
```

Если у вас не хватает прав для завершения процесса, то запустите команду top с помощью sudo. Или выполните команду sudo kill <*идентификатор_процесса*> в другом терминале.

Комментарий

Остановка процесса с помощью команды kill не всегда лучшее решение. Если процесс управляется systemd, то systemd может немедленно перезапустить его, кроме того, от останавливаемого процесса могут зависеть другие процессы,

и тогда есть риск создать беспорядок. Если есть возможность, то остановите процесс с помощью команды `systemctl stop` *<имя_службы>*. Если процесс не управляется systemd, то остановите его с помощью команды `kill`.

Команда `top` всегда по умолчанию предлагает для остановки процесс, находящийся в верхней строчке, то есть использующий больше всего системных ресурсов. Если вам нужно остановить другой процесс, то введите соответствующий идентификатор процесса.

Вы спросите, что такое `sigterm`? Это название сигнала. Сигналы достались в наследство от Unix и за долгие годы обросли многочисленными вариациями. Исчерпывающую информацию о сигналах, таких как `SIGHUP`, `SIGINT`, `SIGQUIT` и других, можно получить в руководстве `man 2 signal`.

Для пользователей и системных администраторов наиболее походящими являются `SIGKILL` и `SIGTERM`. Всегда сначала пробуйте послать сигнал `SIGTERM`, поскольку, получив этот сигнал, процесс попытается завершиться корректно, предварительно передав все дочерние процессы под управление `INIT`, а не оставить их сиротами, и уведомить родительские процессы. Единственный недостаток `SIGTERM` — процесс может игнорировать его.

Используйте `SIGKILL`, только когда `SIGTERM` не дает нужного эффекта. Процессы не могут игнорировать `SIGKILL`, но он также останавливает дочерние процессы, что может повлиять на работу других процессов. Процесс может быть оставлен в подвешенном состоянии как зомби-процесс, поскольку родитель не был уведомлен о его остановке. Зомби-процессы сами по себе не представляют большой проблемы, они просто сидят и ничего не делают. Вы можете увидеть их, если они у вас есть, в заголовке `top` вверху справа в строке `Tasks`. В следующем примере можно видеть, что `top` обнаружила два процесса-зомби:

```
Tasks: 249 total, 1 running, 248 sleeping, 0 stopped, 2 zombie
```

Вам не нужно ничего делать с зомби, поскольку родительское приложение должно автоматически удалить их. Но даже если этого не произойдет, то ничего страшного, разве что появится слишком большое количество зомби. Это говорит о проблеме с приложением. Процессы-зомби нельзя остановить, поскольку они уже стоят. Они потребляют совсем немного системных ресурсов, но если вы захотите избавиться от них, то попробуйте отправить им сигнал `SIGCHLD`:

```
$ sudo kill -s SIGCHLD 1299
```

Если один и тот же процесс продолжает чрезмерно потреблять системные ресурсы, то загляните в его конфигурационный файл с настройками — возможно,

там вы найдете ошибки или сможете настроить программу так, что она будет более эффективно распоряжаться ресурсами. Проверьте также файлы журналов (см. рецепт 20.1); вероятно, там вы найдете подсказки.

Дополнительная информация

- `man 1 top`
- `man 1 kill`

20.9. Обзор выбранных процессов в команде top

Задача

Организовать мониторинг одного или небольшого количества процессов.

Решение

Запустите команду `top` со списком процессов для мониторинга через запятую:

```
$ top -p 4548, 8685, 9348
top - 10:57:39 up 44 min,  2 users,  load average: 0.10, 0.11, 0.21
Tasks:   3 total,   0 running,   3 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.2 us,  0.2 sy,  0.0 ni, 99.6 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :  15691.4 total,  12989.5 free,   1467.4 used,   1234.4 buff/cache
MiB Swap:  15258.0 total,  15258.0 free,      0.0 used.  13601.1 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
   2907 mysql     20   0 1775688  78584  18396 S   0.0   0.5   0:00.22 mysqld
    927 root      20   0 1569764  39072  29320 S   0.0   0.2   0:00.16 libvirtd
    822 root      20   0   11040   6384   4732 S   0.0   0.0   0:00.02 smartd
```

Теперь вы можете следить только за тем, что хотите видеть, обойдясь без необходимости продираться через полчища работающих процессов. Нажмите клавишу со знаком равенства (=), чтобы вернуться к полному списку процессов.

Дополнительная информация

- `man 1 top`
- `man 1 kill`

20.10. Выход из зависшей среды рабочего стола

Задача

В какой-то момент, когда вы счастливо работали, графическая среда рабочего стола вдруг зависла. Указатель мыши движется очень медленно, рывками или вообще не движется.

Решение

Это одна из моих любимых «фишек» в Linux: переход в консоль из графического сеанса. Нажмите Ctrl+Alt+F2 — и окажетесь в текстовой консоли, скрытой под покровом графического сеанса (рис. 20.5).

```
Welcome to openSUSE Leap 15.2 - Kernel 5.3.18-lp152.19-default (tty2).

localhost login:
```

Рис. 20.5. Консоль Linux

Войдите в систему, и после этого сможете выполнить несколько команд, чтобы устранить неполадки. Для начала запустите команду `top`, чтобы отыскать своенравный процесс, нагружающий вашу систему, и остановите его, проверьте файлы журналов, запустите другие диагностические команды, то есть сделайте все, что потребуется. Решив проблему, нажмите Alt+F7, чтобы вернуться в графическую среду. В худшем случае вам придется выключить или перезагрузить компьютер, что в любом случае лучше, чем принудительное выключение через нажатие кнопки питания.

В разных дистрибутивах Linux могут использоваться разные комбинации клавиш. Alt+F7 — традиционная комбинация для возврата в графическую среду. В Fedora используется Alt+F1. Ничего плохого не случится, если вы попробуете их все.

Комментарий

Другой вариант — открыть сеанс SSH с другого компьютера и попытаться вернуть графический рабочий стол в работоспособное состояние.

Для меня нет ничего лучше, чем одновременная доступность текстовой консоли и графической среды.

Принудительное завершение работы не обязательно закончится катастрофой, как раньше, особенно ныне, когда мы используем журналируемые файловые системы, такие как Ext4, XFS или Btrfs.

Стандартная конфигурация — семь консолей, от F1 до F7. Каждая из них является независимым сеансом входа в систему.

Нажмите Ctrl+Alt+Fn, чтобы выйти из графического сеанса и войти в консоль, а находясь в ней, используйте Alt+Fn.

20.11. Устранение неполадок с оборудованием

Задача

У вас есть подозрение в неисправности оборудования, и вам нужно его проверить.

Решение

Если вы полагаете, что оборудование неисправно, сначала попробуйте рецепты из этой главы, описывающие приемы мониторинга оборудования. Некоторые системные прошивки UEFI включают свои мониторы работоспособности оборудования.

Если, опираясь на информацию мониторинга, вы не смогли прийти к какому-то однозначному выводу, то выключите компьютер, откройте корпус, вычистите пыль, очистите фильтры, если они есть, затем разъедините все разъемы: кабели питания, кабели SATA, графические адаптеры и другие карты расширения PCI, модули памяти и разъемы для вентиляторов. Осторожно снова подключите все и обратите особое внимание на правильность установки модулей памяти.



Как не убить свое оборудование или себя

Будьте осторожны! Заземлитесь, прикоснувшись к чему-нибудь, чтобы снять статическое электричество. Наденьте антистатический браслет и кладите извлекаемые компоненты на антистатический коврик. Отключите компьютер от сети и НИКОГДА не прикасайтесь к чему-либо внутри корпуса, когда он включен.

Проверьте блок питания с помощью мультиметра, если знаете, как это сделать, или попробуйте вставить другой блок питания. Пользоваться мультиметром довольно просто и в Интернете можно найти множество инструкций. Если у вас есть запасные части, то замените подозрительные компоненты; возможно, это поможет устранить неисправность.

Закончив и собрав все снова, посмотрите, исчезла ли проблема. В моих компьютерных приключениях проблемы часто решались простой заменой модулей памяти или их переустановкой в разные слоты. Обратите внимание, что на большинстве материнских плат необходимо устанавливать карты с ОЗУ парами в определенные слоты. Некоторые проблемы, связанные с ОЗУ, проявляются как повреждение данных, неполная загрузка и странное поведение, например, когда вы нажимаете выключатель питания для запуска вашей системы, а она не запускается, как бывает при неисправности блока питания.

Убедитесь, что вентиляторы в корпусе ориентированы правильно. Воздух должен втягиваться в корпус обычно спереди и с боков и выбрасываться сзади.

В стране Linux довольно много инструментов для проверки оборудования. Некоторые производители предлагают собственные инструменты для тестирования оборудования и системы; например, Lenovo ThinkPads поставляются со средствами комплексного тестирования, проверяющими все компоненты системы.

GtkStressTesting (https://oreil.ly/7gEST) — отличная утилита для стресс-тестирования процессора, памяти и других компонентов. Она сообщает подробную информацию о материнской плате. Следуйте инструкциям в руководстве по установке, чтобы установить ее в свою систему. В нее входят мониторы, аналогичные lm-sensors.

Единственная функция, которой нет в данной утилите, — мониторинг ввода/вывода, необходимый для выявления узких мест в производительности. Для этой цели используйте утилиту *iotop*, сообщающую информацию о производительности диска в интерфейсе, напоминающем интерфейс *top*.

Комментарий

Иногда сложно отличить программные проблемы от аппаратных. Будьте последовательны и внимательны, поскольку спешка обычно приводит к лишним тратам времени. Используйте доступные справочники для вашего дистрибутива Linux, так как каждый дистрибутив имеет специфические проблемы. Всегда читайте примечания к выпуску.

Дополнительная информация

- `man 8 iotop`
- GtkStressTesting (https://oreil.ly/7gEST).
- Документация с описанием аппаратных компонентов.
- Документация для дистрибутива Linux, форумы, вики-страницы и примечания к выпуску.
- Глава 10.
- Рецепт 20.6.

ГЛАВА 21

Устранение неполадок с сетью

Выявление проблем с сетью почти ничем не отличается от выявления любых других проблем. Вы должны знать параметры настройки своей сети, уметь использовать основные инструменты, проявлять терпение и действовать последовательно и неторопливо.

В этой главе вы узнаете, как использовать *ping*, FPing, Nmap, *httping*, *arping* и *mtr* для проверки подключения, получения карты сети, поиска несанкционированных служб, тестирования производительности сайтов, поиска повторяющихся IP-адресов и маршрутов с низкой пропускной способностью.

Диагностическое оборудование

Если вы застряли перед несколькими неподписанными кабелями с разъемами для подключения к сети Ethernet или к телефонной проводке, то попробуйте протестировать их с помощью тонального пробника. В продаже можно найти множество моделей таких пробников стоимостью меньше 100 долларов. Они состоят из двух частей: тонального генератора и приемника. Два человека, расположившись на разных концах, могут быстро с его помощью выяснить, какой кабель куда идет. Найдя оба конца одного кабеля, подпишите их и двигайтесь дальше. То же самое можно сделать в одиночку, но вдвоем работа пойдет намного быстрее.

Для решения многих задач можно использовать мультиметры, например для обнаружения коротких замыканий и разрывов, проверки целостности и затухания, определения правильности подключения проводов, проверки электрических розеток и блоков питания компьютеров, а также материнских плат.

Adafruit (https://adafruit.com) — отличный сайт, на котором можно найти подробные руководства по использованию мультиметра и обучению электронике.

По возможности имейте под рукой некоторые запасные части. Иногда бывает быстрее заменить сетевой интерфейс, кабель или коммутатор, чтобы искать неисправность.

21.1. Проверка соединения с помощью утилиты ping

Задача

Некоторые службы или хосты в вашей сети недоступны или работают с перебоями. Требуется выяснить, в чем кроется проблема: в оборудовании, службе имен, маршрутизации или в чем-то еще.

Решение

Разбираясь с сетевыми проблемами, начинайте с самого близкого и затем последовательно отходите все дальше и дальше. Имеется в виду и физическое расстояние, и количество маршрутизаторов, которые необходимо пересечь на пути к службе. Начните со своего сегмента локальной сети. Затем переходите к следующему (если у вас их несколько), отделяемому одним маршрутизатором. Затем к следующему, отделяемому двумя маршрутизаторами, и т. д.

Для начала попробуйте старую добрую утилиту *ping*, чтобы проверить соединение. Сначала выполните команду `ping localhost`:

```
$ ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.065 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.035 ms
```

Остановите команду `ping`, нажав Ctrl+C. Проверка localhost позволяет убедиться, что сетевой интерфейс включен и работает. Если появится сообщение connect: Network is unreachable (подключение: Сеть недоступна), значит, проблема в вашем сетевом интерфейсе. Держите под рукой несколько запасных сетевых USB-интерфейсов, чтобы быстро оценить исправность интерфейса.

Убедившись в исправности сетевого интерфейса, выполните команду `ping`, указав имя своего хоста, чтобы проверить разрешение имен, также добавьте параметр, ограничивающий количество посылаемых эхо-запросов:

```
$ ping -c 3 client4
PING client4 (192.168.1.97) 56(84) bytes of data.
64 bytes from client4 (192.168.1.97): icmp_seq=1 ttl=64 time=0.087 ms
64 bytes from client4 (192.168.1.97): icmp_seq=2 ttl=64 time=0.059 ms
64 bytes from client4 (192.168.1.97): icmp_seq=3 ttl=64 time=0.061 ms

--- client4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2046ms
rtt min/avg/max/mdev = 0.059/0.069/0.087/0.012 ms
```

Если возвращается правильный IP-адрес, значит, разрешение имен настроено верно. Если возвращается локальный адрес, например 127.0.1.1, или сообщение Name or service not known (Имя или служба неизвестны), значит, с вашей конфигурацией DNS что-то не так.

Решив проблему с локальным разрешением имен, проверьте связь с одним из хостов в вашей сети, выполнив команду `ping` с его именем. Если она завершится с сообщением Destination host unreachable (Хост назначения недоступен), то попробуйте выполнить ее с его IP-адресом. Если данная команда выполнится успешно, то проверьте свою службу DNS. Если потерпит неудачу с тем же сообщением, то это означает, что имя хоста и его адрес неверны или хост просто выключен.

Если вы не можете получить доступ к каким-то внешним IP-адресам, то ваш сетевой интерфейс, вероятно, исправен и проблема кроется где-то в сети: в кабеле Ethernet, в точке беспроводного доступа или в коммутаторе. Сообщение Network is unreachable (Сеть недоступна) означает, что ваш компьютер не подключен к сети.

Выясняя источник периодических неполадок, запустите команду `ping` на некоторое продолжительное время, например, настройте отправку 500 эхо-запросов с интервалом 2 секунды, чтобы не перегружать хост или свою сеть, и выведите результаты в текстовый файл. Следующая команда будет добавлять новую информацию в конец файла, благодаря чему вы сможете ее останавливать и перезапускать без потери предыдущих результатов:

```
$ ping -c 500 -i 2 server2 >> server2-ping.txt
```

Или используйте `tee`, чтобы выводить результаты одновременно на экран и в файл:

```
$ ping -c 500 -i 2 server2 | tee server2-ping.txt
```

На хосте, подключенном к нескольким сетям, примените команду `ping -i` ***имя-интерфейса***, чтобы явно указать, какой интерфейс использовать.

Комментарий

Не блокируйте ICMP-сообщения *echo-request*, *echo-reply*, *time-exceeded* и *destination-unreachable*. Некоторые администраторы блокируют все ICMP-сообщения в своих брандмауэрах и таким образом поступают ошибочно, поскольку для правильной работы многих сетевых функций требуется беспрепятственное прохождение хотя бы этих четырех сообщений.

Команда `ping` может издавать короткие звуковые сигналы, если передать ей параметр `-a` (audible — «со звуком»), хотя, вероятно, придется немного повозиться, чтобы настроить звук. Раньше в ПК имелись динамики, подключенные непосредственно к материнской плате, и модуль ядра, управляющий данным динамиком, автоматически загружался при загрузке системы. Возможно, вам знаком раздражающий звук, который издает этот динамик, и, может быть, вы даже пытались заставить его воспроизводить музыку.

В настоящее время динамики исчезли из корпусов ПК, и в ноутбуки больше не встраивают пищалки, подключенные к материнской плате. Но большинство материнских плат для ПК по-прежнему их поддерживают, а современная пищалка имеет очень маленький размер (рис. 21.1). Возможно, вам придется ее купить.



Рис. 21.1. Пищалка для материнской платы компьютера

Подключив пищалку, загрузите модуль ядра `pcspkr`, затем убедитесь, что он загрузился:

```
$ sudo modprobe pcspkr
$ lsmod|grep pcspkr
pcspkr                 16384  0
```

А теперь попробуйте извлечь из него звук. Перейдите в обычную консоль, нажав комбинацию Ctrl+Alt+F2, или запустите X-терминал и введите команду `echo` для

воспроизведения ASCII-символа звонка. Все следующие примеры суть одно и то же, просто разные формы представления ASCII-символа с кодом 7:

```
$ echo -e "\a"
$ tput bel
$ echo -e '\007'
```

Можно просто нажать комбинацию Ctrl+G.

Если в графическом терминале не получилось извлечь звук, то проверьте его настройки и включите звуки. И *xfce4-terminal*, и *gnome-terminal* — оба воспроизводят звук при выводе ASCII-символа с кодом 7. *Konsole* поддерживает возможность использования выбранных вами звуковых файлов для уведомлений, но не поддерживает вывод звука на пищалку.

Дополнительная информация

- `man 8 ping`
- Список параметров ICMP, утвержденный организацией IANA (https://oreil.ly/pWYWE).

21.2. Профилирование сети с помощью команды fping и nmap

Задача

Создать список всех хостов и их IP-адресов, находящихся в вашей сети, а также проверить MAC-адреса и открытые порты.

Решение

Используйте команду `fping` и nmap для сканирования локальной сети и запишите результаты.

Команда `fping` последовательно проверяет все адреса в заданном диапазоне. В следующем примере она исследует подсеть, сообщает, какие узлы активны, запрашивает имена узлов в DNS и выводит сводку:

```
$ fping -c1 -gAds 192.168.1.0/24 2>1 | egrep -v "ICMP|xmt" >> fping.txt
client1.net (192.168.1.15)     : [0], 84 bytes, 3.12 ms (3.12 avg, 0% loss)
server2.net (192.168.1.91)     : [0], 84 bytes, 5.34 ms (5.34 avg, 0% loss)
client4.net (192.168.1.97)     : [0], 84 bytes, 0.03 ms (0.03 avg, 0% loss)
```

```
     254 targets
       3 alive
     251 unreachable
       0 unknown addresses

     251 timeouts (waiting for response)

 0.03 ms (min round trip time)
 2.83 ms (avg round trip time)
 5.34 ms (max round trip time)
        3.575 sec (elapsed real time)
```

Чтобы увидеть полный вывод, опустите часть `2>1 | egrep -v "ICMP|xmt"` команды. Выключенные компьютеры не будут обнаружены, поэтому, чтобы составить максимально полный список, может понадобиться запустить приведенную команду несколько раз. Часть `>> fping.txt` добавляет новые результаты в конец при каждом запуске.

Следующий пример с nmap решает аналогичную задачу, но производит менее подробный вывод:

```
$ sudo nmap -sn 192.168.1.0/24 > nmap.txt
Starting Nmap 7.70 ( https://nmap.org ) at 2021-03-31 18:30 PDT
Nmap scan report for client1.net (192.168.1.15)
Host is up (0.0052s latency).
MAC Address: 44:A5:6E:D7:8F:B9 (Unknown)
Nmap scan report for BRW7440BBC7CA75.net (192.168.1.39)
Host is up (1.0s latency).
MAC Address: 74:40:BB:C7:CA:75 (Unknown)
Nmap scan report for client4.net (192.168.1.97)
Host is up (0.47s latency).
MAC Address: 9C:EF:D5:FE:8F:20 (Panda Wireless)
Nmap scan report for server2.net (192.168.1.91)
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 15.19 seconds
```

Информация представлена в довольно удобочитаемом виде, и все же не помешало бы добавить пустые строки перед каждым хостом и сохранить результат в другом файле:

```
$ awk '/Nmap/{print ""}1' nmap.txt > nmap2.txt
```

Теперь благодаря группировке результаты выглядят намного лучше:

```
Nmap scan report for client1.net (192.168.1.15)
Host is up (0.0052s latency).
MAC Address: 44:A5:6E:D7:8F:B9 (Unknown)

Nmap scan report for BRW7440BBC7CA75.net (192.168.1.39)
Host is up (1.0s latency).
MAC Address: 74:40:BB:C7:CA:75 (Unknown)
```

```
Nmap scan report for client4.net (192.168.1.97)
Host is up (0.47s latency).
MAC Address: 9C:EF:D5:FE:8F:20 (Panda Wireless)

Nmap scan report for server2.net (192.168.1.91)
Host is up.

Nmap done: 256 IP addresses (6 hosts up) scanned in 15.19 seconds
```

Следующий пример показывает, как проверить наличие открытых портов на хостах в вашей сети:

```
$ sudo nmap -sS  192.168.1.*
Starting Nmap 7.70 ( https://nmap.org ) at 2021-03-31 19:36 PDT
Nmap scan report for client2.net (192.168.1.15)
Host is up (0.027s latency).
Not shown: 997 closed ports
PORT     STATE    SERVICE
53/tcp   open     domain
80/tcp   open     http
MAC Address: 44:A5:6E:D7:8F:B9 (Unknown)

Nmap scan report for 192.168.1.39
Host is up (0.074s latency).
Not shown: 994 closed ports
PORT     STATE SERVICE
25/tcp   open  smtp
80/tcp   open  http
443/tcp  open  https
515/tcp  open  printer
631/tcp  open  ipp
9100/tcp open  jetdirect
MAC Address: 74:40:BB:C7:CA:75 (Unknown)
[...]
```

На client2.net действуют веб-сервер и сервер DNS. Подобное сканирование можно выполнить извне вашего брандмауэра, чтобы увидеть, доступны ли эти серверы из-за пределов вашей сети.

Обратите внимание на вторую запись — это сетевой принтер, на котором запущено сразу несколько служб. В документации к принтеру говорится, что у каждой из них есть своя цель. Принтер поддерживает удаленное администрирование через веб-панель управления, поэтому при необходимости их можно отключить.

Собрать список хостов и их IP-адресов можно следующим образом:

```
$ nmap -sn 192.168.43.0/24 | grep 'Nmap scan report for' |cut -d' ' -f5,6
server2 (192.168.43.15)
dns-server (192.168.43.74)
client4 (192.168.43.14)
```

Комментарий

nmap поддерживает большое количество параметров для настройки сканирования сетей. Не сканируйте чужие сети без разрешения, поскольку это может быть расценено как акт поиска уязвимостей.

Сканирование портов занимает некоторое время, но его рекомендуется делать регулярно, чтобы видеть, что происходит в сети. Главное правило соблюдения безопасности — запускать только необходимые службы и отключать все остальное.

Дополнительная информация

- `man 1 nmap`
- https://nmap.org.
- `man 8 fping`
- https://fping.org.

21.3. Поиск повторяющихся IP-адресов с помощью утилиты arping

Задача

Определить наличие в сети повторяющихся IP-адресов.

Решение

Ниже представлен пример поиска адреса 192.168.1.91 и отправки по этому адресу четырех эхо-запросов:

```
$ sudo arping -I wlan2 -c 4 192.168.1.91
ARPING 192.168.1.91
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=0 time=49.463 msec
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=1 time=458.306 msec
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=2 time=73.938 msec
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=3 time=504.482 msec

--- 192.168.1.91 statistics ---
4 packets transmitted, 4 packets received, 0% unanswered (0 extra)
rtt min/avg/max/std-dev = 49.463/271.547/504.482/210.659 ms
```

Все MAC-адреса одинаковые, поэтому можно сказать, что в данный момент в сети присутствует только один хост с указанным IP-адресом. А вот пример, когда утилита *arping* обнаружила повторяющийся IP-адрес:

```
$ sudo arping -I wlan2 -c 4 192.168.1.91
ARPING 192.168.1.91
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=0 time=49.463 msec
42 bytes from 2F:EF:D5:FE:8F:20 (192.168.1.91): index=1 time=458.306 msec
42 bytes from 9c:ef:d5:fe:01:7c (192.168.1.91): index=2 time=73.938 msec
42 bytes from 2F:EF:D5:FE:8F:20 (192.168.1.91): index=3 time=504.482 msec
[...]

--- 192.168.1.91 statistics ---
4 packets transmitted, 4 packets received, 0% unanswered (0 extra)
rtt min/avg/max/std-dev = 49.463/271.547/504.482/210.659 ms
```

С помощью nmap можно идентифицировать две машины, использующие один и тот же IP-адрес:

```
$ nmap -sn 192.168.43.0/24 | grep 'Nmap scan report for' |cut -d' ' -f5,6
```

Комментарий

Протокол разрешения адресов (Address Resolution Protocol), *arp*, сопоставляет IP-адреса с MAC-адресами.

Использование DHCP для динамического назначения IP-адресов делает риск появления хостов с одинаковыми IP-адресами меньше, чем при назначении статических IP-адресов вручную. Впрочем, DHCP позволяет также назначать статические адреса, как рассказывалось в главе 16.

Утилита arping помогает увидеть, активен ли хост, когда ping его не находит. Некоторым нравится блокировать ICMP-пакеты, что не очень хорошо, поскольку протокол играет важную роль в работе сети. Утилиту нельзя заблокировать, не отключив возможность сетевых хостов взаимодействовать друг с другом. Протокол разрешения адресов arp поддерживает таблицу MAC-адресов. Когда сетевой хост отправляет пакет другому хосту, arp сопоставляет IP-адрес с MAC-адресом, после чего производится доставка пакета.

Увидеть, как это выглядит, когда arp проверяет сеть, чтобы обновить таблицу адресов, можно с помощью сниффера (перехватчика) пакетов, такого как tcpdump:

```
$ sudo tcpdump -pi eth1 arp
listening on eth1, link-type EN1000MB (Ethernet), capture size 262144 bytes
21:19:36.921293 ARP, Request who-has client4.net tell m1login.net, length 28
21:19:36.921309 ARP, Reply client4.net is-at 9c:ef:d5:fe:8f:20
```

Дополнительная информация

- Глава 16.
- `man 8 arping`

21.4. Проверка пропускной способности и задержки HTTP-сервера с помощью утилиты httping

Задача

Протестировать свой сайт, чтобы убедиться, что его страницы загружаются за разумный интервал времени.

Решение

Утилита *httping* измеряет пропускную способность и задержку ответа HTTP-сервера. Ниже представлен простейший пример оценки задержки:

```
$ httping -c4 -l -g www.oreilly.com
PING www.oreilly.com:443 (/):
connected to 184.86.29.153:443 (453 bytes), seq=0 time=292.25 ms
connected to 184.86.29.153:443 (453 bytes), seq=1 time=726.35 ms
connected to 184.86.29.153:443 (452 bytes), seq=2 time=629.11 ms
connected to 184.86.29.153:443 (453 bytes), seq=3 time=529.95 ms
--- https://www.oreilly.com/ ping statistics ---
4 connects, 4 ok, 0.00% failed, time 6179ms
round-trip min/avg/max = 292.2/544.4/726.3 ms
```

Это не оценка длительности загрузки страниц, а всего лишь оценка времени в миллисекундах, необходимого серверу, чтобы ответить на запрос HEAD, который выбирает только заголовки страниц без содержимого. Запрос `GET` (`-G`) извлекает всю страницу:

```
$ httping -c4 -l -Gg www.oreilly.com
PING www.oreilly.com:443 (/):
connected to 104.112.183.230:443 (453 bytes), seq=0 time=2125.72 ms
connected to 104.112.183.230:443 (453 bytes), seq=1 time=701.94 ms
connected to 104.112.183.230:443 (453 bytes), seq=2 time=470.66 ms
connected to 104.112.183.230:443 (453 bytes), seq=3 time=433.11 ms
--- https://www.oreilly.com/ ping statistics ---
4 connects, 4 ok, 0.00% failed, time 7733ms
round-trip min/avg/max = 433.1/932.9/2125.7 ms
```

Добавьте параметр `-r`, чтобы минимизировать задержки DNS и выполнять разрешение имени хоста только один раз:

```
$ httping -c4 -l -rGg www.oreilly.com
PING www.oreilly.com:443 (/):
```

```
connected to 23.10.2.218:443 (452 bytes), seq=0 time=961.29 ms
connected to 23.10.2.218:443 (452 bytes), seq=1 time=1091.16 ms
connected to 23.10.2.218:443 (452 bytes), seq=2 time=925.46 ms
connected to 23.10.2.218:443 (452 bytes), seq=3 time=913.26 ms
--- https://www.oreilly.com/ ping statistics ---
4 connects, 4 ok, 0.00% failed, time 7894ms
round-trip min/avg/max = 913.3/972.8/1091.2 ms
```

Если минимизация задержек DNS дает существенное улучшение, то это явный сигнал о том, что вам следует обратить внимание на свои серверы имен.

Проверьте альтернативный порт, такой как 8080, добавив его в URL:

```
$ httping -c4 -l -rGg www.oreilly.com:8080
```

Используйте параметр `-s` для вывода возвращаемых кодов состояния, таких как `200 OK`, которые сообщают об успешной загрузке страницы:

```
$ httping -c4 -l -srGg www.oreilly.com
PING www.oreilly.com:443 (/):
connected to 23.10.2.218:443 (452 bytes), seq=0 time=920.88 ms 200 OK
connected to 23.10.2.218:443 (452 bytes), seq=1 time=857.60 ms 200 OK
connected to 23.10.2.218:443 (452 bytes), seq=2 time=1246.69 ms 200 OK
connected to 23.10.2.218:443 (452 bytes), seq=3 time=1134.91 ms 200 OK
--- https://www.oreilly.com/ ping statistics ---
4 connects, 4 ok, 0.00% failed, time 8249ms
round-trip min/avg/max = 857.6/1040.0/1246.7 ms
```

Комментарий

Выполните тестирование несколько раз в разное время дня, чтобы собрать данные, отражающие общую картину, наблюдаемую вашими пользователями.

Утилита httping не является сверхсложным инструментом тестирования, способным углубиться в недра вашего сайта в поисках узких мест, — это простое и быстрое средство, которое поможет получить представление об общей производительности вашего сайта и подскажет, стоит ли тратить время на выявление проблемы с производительностью.

Дополнительная информация

- Коды состояния HTTP (https://oreil.ly/pMvFV).
- `man 1 httping`
- httping (https://oreil.ly/2ts3n).

21.5. Поиск проблемных маршрутизаторов с помощью утилиты mtr

Задача

Получить доступ к сайту, который работает очень медленно или недостижим.

Решение

Используйте *mtr* (my traceroute — «мой трассировщик маршрутов»), чтобы узнать, где пропадают ваши сетевые пакеты. Эта утилита лучше работает в сетях, контролируемых вами, поскольку Интернет обширен и маршруты в нем непостоянны. Но если у вас возникнут проблемы с доступом к сайту, она поможет получить полезную информацию.

Посмотрим, какой из путей приведет нас к `carlaschroder.com`:

```
$ mtr -wo LSRABW carlaschroder.com
Start: 2021-03-31T09:54:17-0700
HOST: client4                              Loss%   Snt   Rcv   Avg  Best  Wrst
  1.|-- m1login.net                          0.0%    10    10  55.5   1.2 199.6
  2.|-- 172.26.96.169                        0.0%    10    10  92.3  29.0 243.6
  3.|-- 172.18.84.60                         0.0%    10    10  84.5  29.3 220.3
  4.|-- 12.249.2.25                          0.0%    10    10  80.7  36.4 215.5
  5.|-- 12.122.146.97                        0.0%    10    10  65.6  34.8 156.6
  6.|-- 12.122.111.33                        0.0%    10    10  49.3  35.5  97.6
  7.|-- cr2.st6wa.ip.att.net                 0.0%    10    10  46.7  35.9  64.0
  8.|-- 12.122.111.109                       0.0%    10    10  57.9  31.4 215.4
  9.|-- 12.122.111.81                        0.0%    10    10  72.3  27.6 231.4
 10.|-- 12.249.133.242                       0.0%    10    10 101.2  31.7 263.1
 11.|-- ae6.cbs01.wb01.sea02.networklayer.com  0.0%    10    10  93.7  31.6 202.7
 12.|-- fc.11.6132.ip4.static.sl-reverse.com  0.0%    10    10 106.0  86.1 171.2
 13.|-- ae1.cbs02.eq01.dal03.networklayer.com 60.0%    10     4 102.0  86.5 115.8
 14.|-- ae0.dar01.dal13.networklayer.com     0.0%    10    10 103.7  80.3 230.8
 15.|-- 85.76.30a9.ip4.static.sl-reverse.com  0.0%    10    10 114.8  82.8 305.7
 16.|-- a1.76.30a9.ip4.static.sl-reverse.com  0.0%    10    10 122.7  83.7 278.4
 17.|-- hs17.name.tools                      0.0%    10    10 145.9  74.9 277.2
```

`m1login.net` — это маршрутизатор шлюза моей сети. За ним находится весь остальной Интернет. Переход под номером 13, скорее всего, является узким местом, поскольку в нем теряется 60 % пакетов. Он может быть частью кластера балансировки нагрузки; на такую мысль наталкивают одинаковые доменные имена переходов 11 и 14. Если это часть кластера, то потеря пакетов несущественна.

Попробуем отправить эхо-запрос последнему переходу — `hs17.name.tools`. Как показывает следующий пример, никаких проблем не возникает:

```
$ ping -c 3 hs17.name.tools
PING hs17.name.tools (169.61.1.230) 56(84) bytes of data.
64 bytes from hs17.name.tools (169.61.1.230): icmp_seq=1 ttl=46 time=319 ms
64 bytes from hs17.name.tools (169.61.1.230): icmp_seq=2 ttl=46 time=168 ms
64 bytes from hs17.name.tools (169.61.1.230): icmp_seq=3 ttl=46 time=166 ms
[...]
```

Если `mtr` обнаружит проблему, то используйте `whois` для поиска владельца домена и его контактной информации:

```
$ whois -H networklayer.com
```

`whois` также работает с IP-адресами. Параметр `-H` отключает вывод надоедливых юридических предупреждений.

Сохраните вывод `mtr` в файл с датой и временем в конце каждой записи:

```
$ mtr -r -c25 oreilly.com >> mtr.txt && date >> mtr.txt
```

Организуйте сбор данных с течением времени, создав задание cron (см. рецепт 3.7), которое запускает предыдущую команду `mtr` раз в час, и пусть это задание поработает день-другой. И не забудьте отключить его!

Комментарий

`mtr -wo LSRABW` ограничивает количество столбцов, чтобы данные лучше помещались на этой странице. `mtr -w` — это широкий формат вывода для отчетов.

Сохраните полученные данные на случай, если вам захочется сообщить о проблеме; в выводе `whois` описывается, как найти того, к кому можно обратиться.

`mtr` генерирует довольно объемный трафик, поэтому не запускайте ее слишком часто.

Дополнительная информация

- `man 8 mtr`

ПРИЛОЖЕНИЕ

Шпаргалки по управлению программным обеспечением

Программное обеспечение для Linux распространяется в *пакетах*. Они содержат все файлы, принадлежащие определенному приложению, например браузеру, текстовому процессору или игре. В системах Linux широко распространены разделяемые библиотеки, которые совместно используются несколькими приложениями. Большинство пакетов в Linux не являются полностью самодостаточными и зависят от разделяемых файлов.

В роли диспетчера ПО в большинстве дистрибутивов Linux используется утилита GNOME-Software (Программное обеспечение Gnome), или просто Software (рис. П.1). Она имеет хорошо организованный графический интерфейс с категориями и богатыми возможностями поиска.

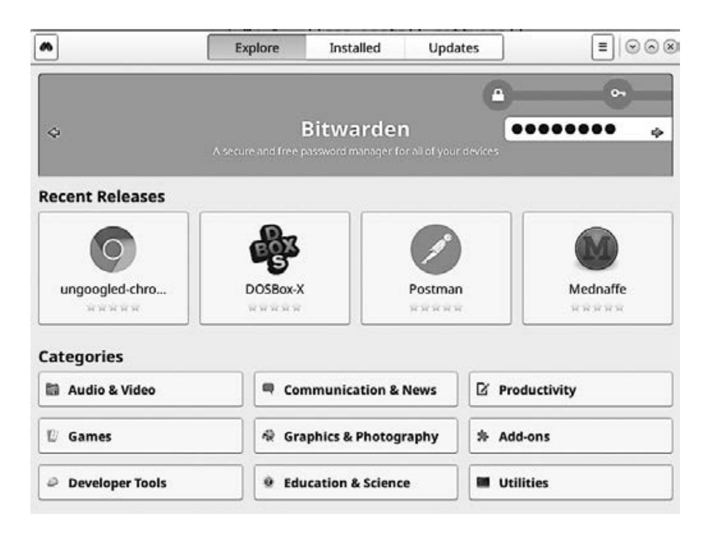


Рис. П.1. GNOME-Software

Команды управления пакетами

В любом дистрибутиве Linux есть три типа команд управления программным обеспечением.

- Диспетчер пакетов, который управляет только отдельными пакетами. В Fedora и openSUSE используется диспетчер пакетов `rpm`, в Ubuntu — `dpkg`.
- Диспетчер пакетов с поддержкой разрешения зависимостей. В Fedora используется `dnf`, в openSUSE — `zypper`, а в Ubuntu — `apt`. Диспетчеры пакетов с поддержкой разрешения зависимостей обеспечивают автоматическую установку любых зависимостей для конкретного пакета. Например, текстовый редактор gedit имеет длинный список зависимостей, как показано в следующем примере для `apt`:

  ```
  $ apt depends gedit
  gedit
    Depends: gedit-common (<< 3.37)
    Depends: gedit-common (>= 3.36)
    Depends: gir1.2-glib-2.0
    Depends: gir1.2-gtk-3.0 (>= 3.21.3)
    Depends: gir1.2-gtksource-4
    Depends: gir1.2-pango-1.0
    Depends: gir1.2-peas-1.0
    Depends: gsettings-desktop-schemas
    Depends: iso-codes
  [...]
  ```

  Управлять зависимостями вручную сложно; диспетчеры пакетов с поддержкой разрешения зависимостей многократно упрощают жизнь пользователям Linux.
- Команды для управления группами взаимосвязанных пакетов, такими как графическая среда рабочего стола, звук и видео или серверные стеки. В openSUSE подобные группы называются шаблонами, в Fedora — группами пакетов, в Ubuntu — задачами. В следующем примере показаны некоторые шаблоны openSUSE:

  ```
  $ zypper search --type pattern
  S  | Name                  | Summary                         | Type
  ---+-----------------------+---------------------------------+--
  [...]
     | mail_server           | Mail and News Server            | pattern
     | mate                  | MATE Desktop Environment        | pattern
  i+ | multimedia            | Multimedia                      | pattern
     | network_admin         | Network Administration          | pattern
     | non_oss               | Misc. Proprietary Packages      | pattern
     | office                | Office Software                 | pattern
     | print_server          | Print Server                    | pattern
  [...]
  ```

Пакеты программного обеспечения распространяются через репозитории — общедоступные серверы, откуда все мы скачиваем пакеты. Вы можете посетить их при желании:

- репозитории Fedora (https://oreil.ly/nLDaM);
- репозитории openSUSE (https://oreil.ly/H8clz);
- поиск пакетов для Ubuntu (https://oreil.ly/BZw5d).

У каждого дистрибутива Linux есть свои официальные репозитории, а также множество сторонних репозиториев. В этом приложении описаны основные команды для управления программным обеспечением и репозиториями в системе Linux.

Управление программным обеспечением в Ubuntu

В этой книге Ubuntu Linux представляет целое семейство дистрибутивов на основе Debian. Сначала был Debian, затем появились сотни производных от него. Основные наследники Debian используют одну и ту же систему управления пакетами, и команды, описываемые в данном приложении, должны во всех них работать одинаково.

В этом приложении представлены три команды управления программным обеспечением: `dpkg`, `apt` и `tasksel`.

Использование add-apt для подключения и отключения репозиториев

При подключении репозитория с программным обеспечением вам понадобится указать кодовое название вашей версии Ubuntu. Узнать его можно с помощью следующей команды:

```
$ lsb_release -sc
focal
```

Вам нужно задать точный URL репозитория, предоставленный теми, кто поддерживает репозиторий:

```
$ sudo add-apt-repository "deb http://us.archive.ubuntu.com/ubuntu/ focal \
universe multiverse"
```

Отключить репозиторий можно так:

```
$ sudo add-apt-repository -r "deb http://us.archive.ubuntu.com/ubuntu/ focal \
universe multiverse"
```

После подключения или отключения репозитория не забудьте обновить кэш пакетов:

```
$ sudo apt update
```

Выполняйте эту команду регулярно для скачивания обновлений из репозитория. Затем установите обновления:

```
$ sudo apt upgrade
```

Установка, удаление и исследование содержимого пакетов с помощью dpkg

Как рассказывалось в разделе «Команды управления пакетами» на с. 581, утилита dpkg оперирует отдельными пакетами и не поддерживает разрешение зависимостей.

Установка пакета:

```
$ sudo dpkg -i имя_пакета
```

Удаление пакета (без удаления конфигурационных файлов):

```
$ sudo dpkg -r имя_пакета
```

Удаление пакета с конфигурационными файлами:

```
$ sudo dpkg --purge имя_пакета
```

Список содержимого пакета:

```
$ dpkg -L имя_пакета
```

Список всех установленных пакетов:

```
$ dpkg-query --listdpkg
```

Установка, удаление, поиск и исследование содержимого пакетов с помощью apt

`apt` — это диспетчер пакетов с поддержкой разрешения зависимостей, ваша повседневная команда управления программным обеспечением.

Поиск пакета:

```
$ apt search имя_пакета
```

Поиск искомой строки только в названиях пакетов:

```
$ apt search имя_пакета --names-only
```

Вывод подробной информации о пакете:

```
$ apt show имя_пакета
```

Установка пакета:

```
$ sudo apt install имя_пакета
```

Удаление пакета (без удаления конфигурационных файлов):

```
$ sudo apt remove имя_пакета
```

Удаление пакета с конфигурационными файлами:

```
$ sudo apt remove purge имя_пакета
```

Использование tasksel

`tasksel` управляет задачами, то есть группами пакетов.

Список доступных задач:

```
$ tasksel --list-tasks
```

Установка задачи:

```
$ sudo tasksel install имя_задачи
```

Удаление задачи:

```
$ sudo tasksel remove имя_задачи
```

Управление программным обеспечением в Fedora

В этой книге Fedora Linux представляет семейство дистрибутивов, основанных на Red Hat Linux. Red Hat, CentOS, Scientific Linux, Oracle Linux и многие другие используют одну и ту же систему управления пакетами, и команды, описываемые в данном приложении, должны во всех них работать одинаково.

В этом приложении представлены две команды управления программным обеспечением: `rpm` и `dnf`.

Управление репозиториями с помощью команды dnf

Список всех подключенных репозиториев, активных и неактивных:

```
$ dnf repolist --all
```

Список активных репозиториев:

```
$ dnf repolist --enabled
```

Вывод подробной информации о репозиториях:

```
$ dnf repolist --enabled
```

Подключение репозитория:

```
$ sudo dnf config-manager --add-repo /etc/yum.repos.d/fedora_extras.repo
```

Активация репозитория:

```
$ sudo dnf config-manager --set-enabled fedora-extras
```

Деактивация репозитория:

```
$ sudo dnf config-manager --set-disabled fedora-extras
```

Управление программным обеспечением с помощью команды dnf

Поиск пакета:

```
$ dnf search имя_пакета
```

Установка пакета:

```
$ sudo dnf install имя_пакета
```

Удаление пакета:

```
$ sudo dnf remove имя_пакета
```

Получение информации о пакете:

```
$ dnf info имя_пакета
```

Установка обновлений:

```
$ sudo dnf upgrade
```

Получение списка групп пакетов:

```
$ dnf grouplist
```

Установка группы пакетов:

```
$ sudo dnf groupinstall "имя_группы"
```

Удаление группы пакетов:

```
$ sudo dnf groupremove "имя_группы"
```

Установка и удаление пакетов с помощью команды rpm

Установка пакета:

```
$ sudo rpm -i имя_пакета
```

Обновление пакета:

```
$ sudo rpm -U имя_пакета
```

Удаление пакета:

```
$ sudo rpm -e имя_пакета
```

Получение информации о пакетах с помощью команды rpm

Список всех файлов в установленном rpm-пакете:

```
$ rpm -ql имя_пакета
```

Вывод полной информации об установленном пакете:

```
$ rpm -qi имя_пакета
```

Просмотр журнала изменений для пакета:

```
$ rpm -q --changes имя_пакета
```

Управление программным обеспечением в openSUSE

В openSUSE используются пакеты в формате RPM, как в Fedora, но управление зависимостями осуществляется с помощью другого диспетчера — `zypper`.

Управление репозиториями с помощью zypper

Список всех подключенных репозиториев:

```
$ zypper repos
```

Список всех подключенных репозиториев с их адресами URL:

```
$ zypper repos -d
```

Активация репозитория:

```
$ sudo zypper modifyrepo -e имя_репозитория
```

Деактивация репозитория:

```
$ sudo zypper modifyrepo -d имя_репозитория
```

Подключение нового репозитория:

```
$ sudo zypper adderepo -name "имя_нового_репозитория" \
http://download.opensuse.org/distribution/leap/15.3/repo/oss/
```

Отключение репозитория:

```
$ sudo zypper removerepo имя_репозитория
```

Скачивание обновлений из репозиториев:

```
$ sudo zypper refresh
```

Управление программным обеспечением с помощью диспетчера zypper

Обновление системы (предварительно выполните команду `sudo zypper refresh`):

```
$ sudo zypper update
```

Поиск пакета (неточный):

```
$ zypper search имя_пакета
```

Поиск пакета (точный):

```
$ zypper search -x имя_пакета
```

Установка пакета:

```
$ sudo zypper install имя_пакета
```

Удаление пакета:

```
$ sudo zypper remove имя_пакета
```

Список всех шаблонов:

```
$ sudo zypper -t patterns
```

Установка шаблона:

```
$ sudo zypper -t pattern имя_шаблона
```

Об авторе

Карла Шрёдер (Carla Schroder) впервые села за компьютер в середине 1990-х годов. За годы, прошедшие с той поры, она работала системным и сетевым администратором в смешанных сетях Linux/Microsoft/Apple, журналистом и техническим писателем. Карла написала более 1000 руководств по Linux для различных изданий и в настоящее время пишет и поддерживает руководства для компании, производящей программное обеспечение для Linux корпоративного уровня. Она является автором книг *Linux Cookbook* (O'Reilly), *Linux Networking Cookbook* (O'Reilly) и *The Book of Audacity* (No Starch Press).

Об обложке

На обложке книги изображен рябчик *(Tetrastes bonasia)*. Эта оседлая птица, которую иногда называют орешниковой курицей, является одним из самых мелких представителей тетеревиных. Ее можно встретить в густых лесах на большей части Восточной Европы и Северной Азии.

Эта пернатая дичь имеет пестрое оперение, более серое в нижней части и более коричневое на крыльях и спине. Самцы рябчиков имеют на голове хорошо выраженный хохолок и черное оперение на горле с белой каймой. Самки имеют более короткий хохолок и коричневое оперение на горле. Основу рациона рябчика составляет растительная пища, но в период размножения в пищу употребляются также насекомые. Самки высиживают яйца и самостоятельно ухаживают за птенцами.

В настоящее время рябчику присвоен природоохранный статус «вызывающий наименьшее беспокойство», но многие животные, изображаемые на обложках книг издательства O'Reilly, находятся под угрозой исчезновения; они все важны для нашего мира.

Иллюстрацию для обложки нарисовала Карен Монтгомери (Karen Montgomery) на основе черно-белой гравюры из энциклопедии *Meyers Kleines Lexicon*.

Карла Шрёдер

Linux. Книга рецептов

2-е издание

Перевел с английского *А. Киселев*

Руководитель дивизиона	*Ю. Сергиенко*
Руководитель проекта	*А. Питиримов*
Ведущий редактор	*Н. Гринчик*
Литературные редакторы	*Н. Рощина, Н. Хлебина*
Художественный редактор	*В. Мостипан*
Корректор	*Е. Павлович*
Верстка	*Г. Блинов*

Изготовлено в России. Изготовитель: ООО «Прогресс книга».
Место нахождения и фактический адрес: 194044, Россия, г. Санкт-Петербург,
Б. Сампсониевский пр., д. 29А, пом. 52. Тел.: +78127037373.

Дата изготовления: 04.2022. Наименование: книжная продукция. Срок годности: не ограничен.

Налоговая льгота — общероссийский классификатор продукции ОК 034-2014, 58.11.12 — Книги печатные профессиональные, технические и научные.

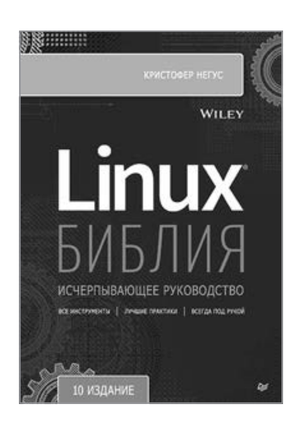
Импортер в Беларусь: ООО «ПИТЕР М», 220020, РБ, г. Минск, ул. Тимирязева, д. 121/3, к. 214, тел./факс: 208 80 01.

Подписано в печать 25.02.22. Формат 70×100/16. Бумага офсетная. Усл. п. л. 47,730. Тираж 1000. Заказ 0000.

Кристофер Негус

БИБЛИЯ LINUX

10-е издание



Полностью обновленное 10-е издание «Библии Linux» поможет как начинающим, так и опытным пользователям приобрести знания и навыки, которые выведут на новый уровень владения Linux. Известный эксперт и автор бестселлеров Кристофер Негус делает акцент на инструментах командной строки и новейших версиях Red Hat Enterprise Linux, Fedora и Ubuntu. Шаг за шагом на подробных примерах и упражнениях вы досконально поймете операционную систему Linux и пустите знания в дело. Кроме того, в 10-м издании содержатся материалы для подготовки к экзаменам на различные сертификаты по Linux.

КУПИТЬ